A FRAMEWORK FOR THE DESIGN OF A MEDICAL TUTORING SYSTEM FOR THE INSTRUCTION OF UNDERGRADUATES IN GENERAL PRACTICE

ALI ABDUL HADI MANSOUR

Doctor of Philosophy in Computer Science

University of Sheffield

Department of Computer Science

Submitted in November 1989

Degree Accpeted in 1990

CONTENTS

CHAPTER IV

DESIGN CONSIDERATIONS FOR A COMPUTER-BASED PRIMARY CARE LEARNING ENVIRONMENT 52

CHAPTER V

SYSTEM IMPLEMENTATION 77

CHAPTER VII

CONCLUSIONS 120

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

A number of people have helped in assessment and evaluation of our tutoring system. First, I would like to express my gratitude to Dr. David Ingram and his colleagues at St. Bartholomew's Hospital Medical College for making an independent evaluation of our tutoring system. I would also like to thank all the students for their valuable comments and suggestions about the system.

TRADEMARKS

Acorn Electron is a trademark of Acorn Computers Limited.

Amstrad is a trademark of Amstrad PLC.

BBC B/MASTER/BASIC are trademarks of the British Broadcasting Corporation.

CP/M is a trademark of Digital Research, Inc.

DEMO is a trademark of Software Garden, Inc.

IBM PC/XT is a registered trademark of International Business Machines, Inc.

MSDOS is a trademark of Microsoft, Corp.

Turbo Prolog is a trademark of Borland International.

A FRAMEWORK FOR THE DESIGN OF A MEDICAL TUTORING SYSTEM FOR THE INSTRUCTION OF UNDERGRADUATES IN GENERAL PRACTICE

ALI ABDUL HADI MANSOUR

SUMMARY

One of the difficulties in teaching clinical medicine is the lack of opportunity a student has to acquire techniques for solving clinical problems. By using a computer to simulate a General Practice environment where patients with sets of symptoms are presented, a student can gain experience of diagnostic techniques and treatment management for any medical condition. Such an approach should enhance a student's development of properly structured clinical algorithms for interrogating a patient and arriving at an appropriate management plan.

The intelligent tutoring system developed at the Department of Computer Science with the collaboration of the Department of General Practice aims not only to simulate this environment but also to formulate the basis for a general interactive learning environment for all subject domains with similar problem-solving model.

In this system, a student may question, examine and provide treatment plans for a patient whilst constantly being monitored by the system. Using Artificial Intelligence techniques, the tutor is able to assess the progress of a student throughout the tutorial session and produce tutoring interventions at appropriate stages, according to the student's ability.

The system's knowledge base consists of disease profiles and population parameters which are created and updated by a separate system - the Medical Editor. The manipulation of this database allows tailoring of the system to simulate any clinical situation in Primary Care.

This research considers in detail the current teaching/tutoring strategies adopted by all medical computer-assisted learning systems. It identifies the main areas of difficulty for using such systems in the Primary Care undergraduate course and discusses the consultation model used in this system with full comparison of the models used in Secondary Care. The research also discusses the main design issues which forms the framework for building learning environments based on intelligent tutoring systems.

CHAPTER I

INTRODUCTION

1.1 OBJECTIVES AND DESIGN OVERVIEW

This research is a joint project with a CTISS (Computers in Teaching Initiative Support Service) project carried out in the Department of Computer Science with the collaboration of the Department of General Practice at the University of Sheffield. The CTISS project aims at the development of a Computer-Assisted Learning (CAL) system for teaching undergraduate students of General Practice. The application of CAL systems in medical teaching can be difficult to justify on an economic basis but in medicine with the high cost of training a doctor, this type of teaching can be cost-effective. Once developed, the CAL system will be distributed to all medical schools with a General Practice department in this country for use, hopefully, as part of their undergraduate teaching curriculum.

The research project focuses attention on computer-based teaching systems for the Primary Care environment*, an environment that has been allocated secondary importance by educationalists from both the medical and computer science fields.

The primary objective of the research project is to provide an outline for the design of a computer-based learning environment for use by students of General Practice. The design of such an environment follows the general consultation model adopted by General Practitioners in their daily work. This model is discussed in detail in Chapter 3.

The project also aims to extend the CAL system to include some Artificial Intelligence (AI) techniques for the development of an Intelligent Tutoring System (ITS). The features to be included in the ITS include:

* 'Primary Care' is another term for 'General Practice'.

- A limited mixed-initiative dialogue system between the tutor and the student.

- A model of the student's knowledge in medical disease diagnosis and management.

The combined result, or more specifically, the final product of the joint project is a complete Medical Tutoring system consisting of a collection of three sub-systems:

- The Medical Data Entry System
- The Computer-Assisted Learning System
- The Intelligent Tutoring System

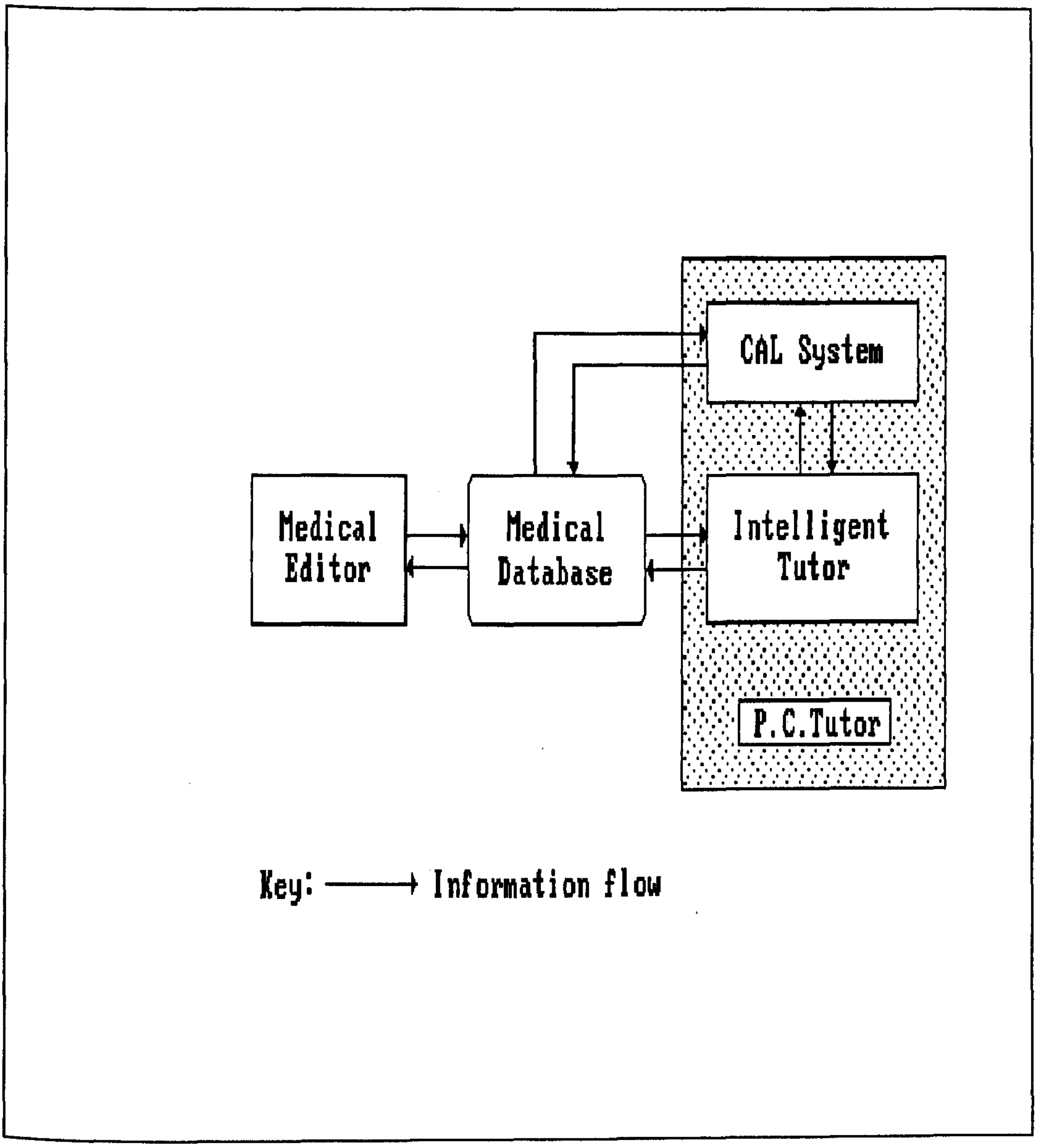


Figure 1.1 - The Sheffield University Medical Tutoring System

Figure 1.1 shows the top-level architecture of the complete medical system. The system is implemented in Prolog running on an IBM PC/XT (TM) compatible machine with 640K RAM. The medical data was supplied by a General Practitioner assigned to the CTISS project. The Medical Data Entry System was designed by myself in collaboration with a research assistant who did most of the coding for this part of the tutoring system. The CAL system was designed in collaboration with the General Practitioner assigned to the CTISS project and implemented by myself. While the intelligent tutor was designed and programmed by myself.

1.2 THE MOTIVATION FOR THE CTISS PROJECT AND RESEARCH

This research and CTISS project are focused on two areas; Medical and Computer Science. From the medical side, both projects are motivated by the use of computers as part of the undergraduate medical curriculum. Despite the fact that computers are getting cheaper and the increasing interest of the government in promoting Information Technology in education, medical teaching systems are yet to be seen in this country. The majority of systems evolved from research projects in medical education have not moved towards their use in medical school curriculums. This is mainly due to the lack of funds and enthusiasm from the educationalists in the field of medicine towards the use of computers in medical education. In addition, the majority of the medical systems that evolved from research projects were implemented on expensive research workstations or old mainframes hence failing to be used outside their place of origin.

This view brings us to the most important motivational factor from the medical side of both projects namely the lack of tutoring systems for the General Practice environment. Computer-based systems for the General Practice environment are limited to administrative procedures only. These systems are designed for use in General Practice surgeries. Their uses include patient records-keeping, drug details, surgery audit procedures, the generation of repeated prescriptions, the automatic generation of personalized letters calling patients due for immunization and/or screening, and so on. Existing teaching and tutoring systems for use by General Practitioners are usually developed and supplied by pharmaceutical companies. Being commercially funded, these systems aim to "teach" practising GPs the correct use of the companies' pharmaceutical products when managing their patients. The rest of medical teaching, tutoring and even expert systems that are available either commercially or developed at medical schools, usually for in-house use, are designed for use in the hospital environment only - an environment which is somewhat different to the General Practice environment as we shall see in Chapter 3. One expert or more precisely an assistant system aimed at the Primary Care environment is the Oxford System of Medicine [Fox *et al.*, 1987]. This is a prototype developed as part of a case study by the Imperial Cancer Research Fund Laboratories into the use of this type of system in General Practice surgeries. This system is still in its prototyping stage.

Another motivation is the almost complete absence of consistent and reproducible mechanisms for student assessment in terms of performance or progress from any medical school curriculum. Indeed there is little information about the *Process* of undergraduate or postgraduate education in medicine and far less understanding about assessment [Kemmis *et al.*, 1977]. For this, the tutor is expected to act as a testbed for the development of a standard from which a comprehensive and consistent protocol for the assessment of a student's educational needs might be produced, and indeed could be further extended to allow any Medical Practitioner to use it as an educational prescription for continuing medical education.

From the computer science viewpoint one of the motivational factors for the research project is the desire to build an interactive tutoring system suitable for use in other domains similar in concept to that of General Practice. Although our tutoring system has the ability to be applied to other domains, no attempt was made to use it in any other domain due to the limited time available for both projects. Examples of such domains include automobile maintenance training, hardware maintenance training, and so on. The application of our system in such domains is considered in Chapter 7. The chapter also includes an example of applying the tutoring system to computer hardware maintenance.

The second motivation and probably the most important and feasible from the computer science viewpoint is to offer students a self-contained interactive learning environment which complements lectures and student readings. What makes it feasible is the availability of high performance personal computers (computers based on RISC, Transputer and other 32-bit processor architectures), and the increasing use of interactive video (IV). IV is a CAL that makes use of visual media (e.g. VideoDisk, CD-ROM, etc.). Such hardware offers researchers and system developers an opportunity to develop high level interactive learning environments for use in all fields. This particular type of environment is valuable from a medical viewpoint in that a student can experiment and extend his or her knowledge and skills of observation without inconveniencing or even risking the lives of human patients. In fact, IV systems are currently being developed in a number of universities in the U.K. for teaching specific areas of medicine (e.g. pathology, dentistry, etc.) [Computers in Medical Education, 1988].

1.3 GENERAL STRUCTURE OF THE TUTORING SYSTEM

B. Woolf [1988] has pointed out that it is important to clarify the teaching philosophy early in the development process. A number of factors guided the choice in our tutoring system. An obvious factor is to minimize long tutor presentations and maximize student involvement. This is an important factor in simulation-based systems. We also wanted to minimize tutor guidance. This ruled out the *Socratic Diagnosis* strategy. This strategy has been much used in tutoring systems. In brief, Socratic Diagnosis relies on the student already having some model of the domain which he or she is learning. Such a model may be flawed in some respects, embodying one or more misconceptions about the nature of the domain. The tutor identifies these misconceptions and then takes the student through a series of educational interactions designed to make the student realize his or her misconceptions.

Another teaching technique was also ruled out. This technique is *Cognitive Apprenticeship* [Collins *et al.*, 1987]. This is based on the idea that cognitive skills can be learnt in the way that crafts are learnt from an expert in the craft. Although suitable for domains such as electronic fault-finding, mathematics, etc., it is not appropriate to the medical domain.

A third factor in our design is concerned with the purpose of the system. Our intention was to build a system that complements or reinforces what the student has learned from lectures, text-books and clinical training. This factor, therefore, discounts the *Discovery Learning* technique.

The teaching strategy which was arrived at for use in our tutoring system with the prototype name *Primary Care Tutor* (*P.C.Tutor* for short) is in fact a combination of two known strategies: *Practice* and *Direct Assessment* [Elsom-Cook *et al.*, 1988]. In general, the *Practice* strategy involves presenting the student with a problem and asking him or her to carry out some task. While the student is engaged in this activity the tutor monitors the student actions in the background. This is achieved with a technique very similar to Anderson's model tracing [Anderson *et al.*, 1986]. The student modelling component observes the student problem-solving activity step by step. If the student deviates from the ideal model (i.e. that of the human expert) the tutor immediately provides remedial feedback. The model built by the tutor will represent the student's knowledge and his or her misconceptions. In *P.C.Tutor*, the Practice strategy has been slightly modified to minimize interruptions. These have, therefore, been limited to incorrect diseases. Implementation details are discussed in Chapter 5. *Direct Assessment* on the other hand, allows for the assessment of students declarative knowledge using the traditional question and answer format. Some of the questions used by this strategy are entered by the teachers while others are automatically generated from the tutorial text. In *P.C.Tutor*, all questions are automatically generated from case data.

P.C.Tutor has the general structure shown in Figure 1.2. The tutor selection of diseases is dependent upon the number of diseases available in the medical database and the consultation profile. The consultation profile represents the number of males and females visiting the simulated surgery in a particular community and their age distribution. This provides the teacher with an opportunity to simulate any environment for his or her students to practise on. The consultation profile used in our system was obtained from a real-life surgery in Sheffield. This information is not confidential and is available to the Family Practitioner Committees which uses such information for its surveys, calling patients for tests, etc.

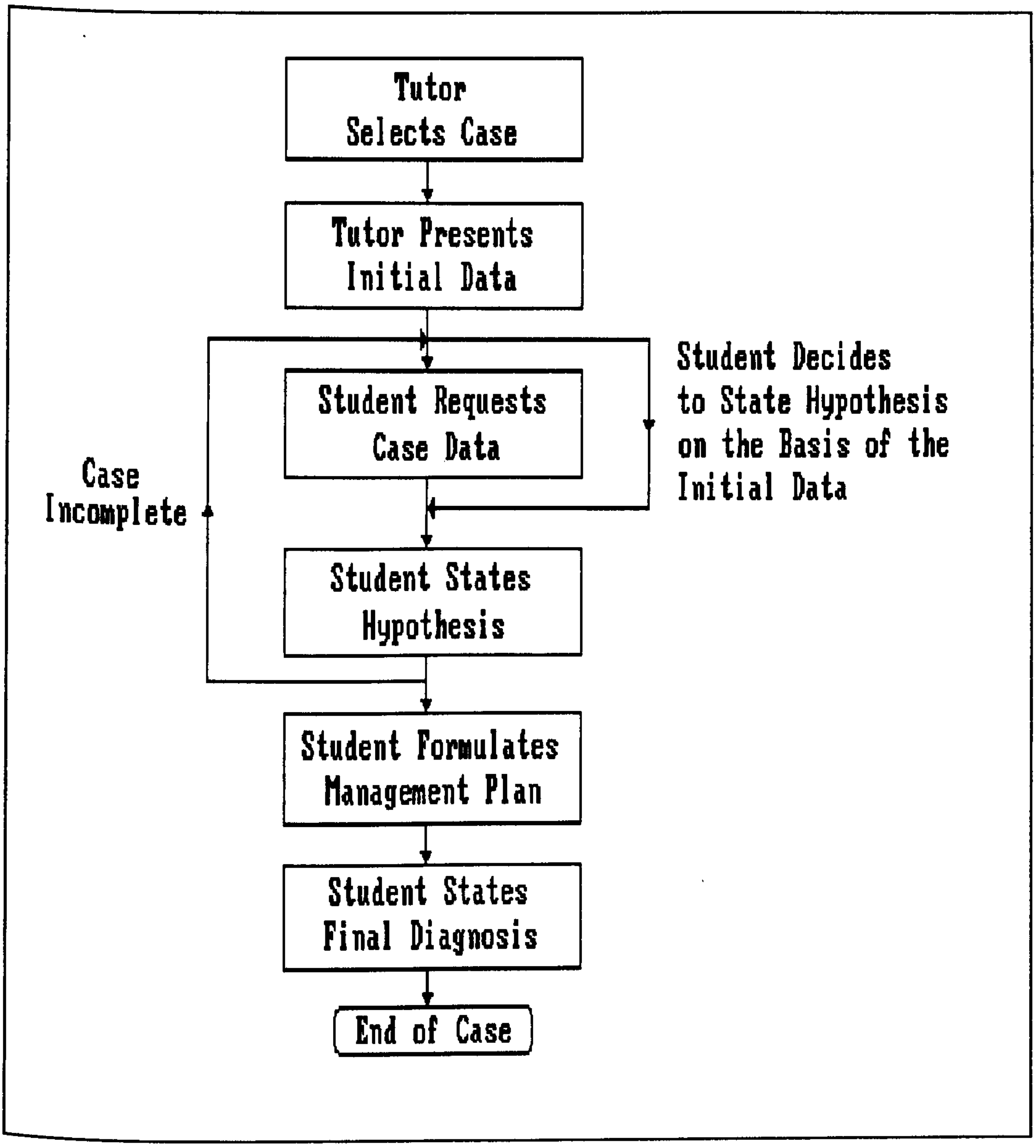


Figure 1.2 - General Structure of a P.C.Tutor Tutorial

P.C.Tutor dialogue is based on all the case information that has been given to the student during a tutorial session. The tutor also takes into account the student overall level of understanding in his or her previous cases. This is commonly known as the *Student Model.*

The student model in *P.C.Tutor* is a numeric overlay model [Carr *et al.*, 1977]. That is, a number of components in the system have numeric values associated with them which are intended to represent the system's beliefs about the knowledge of the student. In an overlay model, the student is represented as a subset of the total knowledge within the system. The scale used is bipolar (i.e. varying from "confident the student does not know this" to "confident the student knows this"). These numeric values are associated with basic actions (e.g. requesting appropriate laboratory tests), goals (both in particular contexts and in general) and concepts in the underlaying domain. This modelling technique is simple and well understood. Details of the student model are in Chapters 4 and 5.

The general structure shown in Figure 1.2 enables the tutor to be used in other domains where the problem-solving concept is similar to that of General Practice tutoring.

The complete medical tutoring system has been designed so that given a complete medical database and a set of facts, an appropriate list of plausible diseases can be generated and ranked according to disease prevalence. This feature is currently being considered for the development of an assistant system based on our medical database. Details of proposed future developments are discussed in Chapter 7.

1.4 OVERVIEW OF THE THESIS*

Although the tutoring system built in this joint project is specific to Primary Care, Chapter 2 examines the teaching of undergraduates in medical schools. I will also consider the appropriateness of using computers in medical teaching in general.

In Chapter 3, I will start by outlining the type of teaching exercises called Patient Management Problems. This will be followed by a discussion of the main differences between General Practice work and Hospital work. Such differences are the main influencing factors in the development of our tutoring system. I will also be reviewing some of the documented medical teaching systems. I will do so by first concentrating on Primary Care systems and then Hospital-based systems as the two types are different in a number of areas. At the end of the chapter, I will be outlining the general structure adopted by most tutoring systems.

Chapter 4 provides an in-depth examination of the design methodology adopted in this research. The implementation of the tutoring system is discussed in Chapter 5.

Chapter 6 considers the results obtained from testing the system by the end users (i.e. the undergraduate medical students). Additional views from hospital consultants and practising General Practitioners are also included.

The conclusions of this research are made in Chapter 7 with some comments on the current state of the joint project and on proposed future development for the complete medical system.

---

* Extracts from this thesis have been published in: (1) the Department of Computer Science (University of Sheffield) research report CS-89-28 titled "Primary Care Tutor: A Medical Tutoring System for Students of General Practice", and (2) the paper "An Intelligent Tutoring System for the Instruction of Medical Students in Techniques of General Practice", co-authored by J.Poyser, J.J.McGregor and M.E.Franklin, in the *Computers and Education Journal, Vol. 15, No.1-3, pp 83-90,* Computer Assisted Learning: Selected Papers from the Proceedings of the CAL 89 Symposium, (1990) M.R.Kibby (Editor), Pergamon Press.

CHAPTER II

COMPUTERS IN MEDICAL EDUCATION

2.1 UNDERGRADUATE TEACHING IN MEDICAL SCHOOLS

The teaching of undergraduates in medical schools has always been designed for hospital-based medicine. A medical student throughout his or her long apprenticeship of clinical teaching must have the opportunity and develop the capability to do several tasks at once; i.e. he or she must deploy clinical skills of observation and communication, use many specialized instruments (e.g. stethoscopes, etc.), and interpret visual data such as ECGs, blood films, X-Rays, and so on.

There are then, three categories or domains in which the student must acquire competence: *interpersonal*, *manipulative*, and *cognitive* [Feinstein, 1977].

The focus of most medical Computer-Assisted Instruction (CAI) systems is the application of computer-based techniques to teaching cognitive skills involved in clinical problem solving. Manipulative skills involve the acquisition of data and treatment by instrumentation. Further, computer-based techniques demonstrating interpersonal attributes or skills are highly unlikely to have any significant impact on undergraduate teaching.

The cognitive domain is divided into three levels of behaviour: (1) recall or recognition which is an effort of memory, (2) interpretation of clinical data, e.g. aetiology of diseases, properties of diseases, etc., and (3) the application of knowledge to problem solving and evaluation.

The cognitive components of medical practice are of fundamental importance since they comprise judgmental knowledge for managing patient problems. Inherent in each patient encounter is the need to identify the aetiology, the correct streamlined differential diagnosis and the choice of appropriate therapy. As Feinstein [1967] has emphasized, care of a patient is a therapeutic experiment, and the outcome depends strongly on the adequacy of the clinician's "clinical judgment" involved in the design, execution, and appraisal of the experiment. Barnett [1974] points out that:

> "Despite the importance of clinical judgment in medical practice, clinical training consists primarily of an apprenticeship in which there is little systematic attempt to teach any explicit model of the intellectual processes involved in medical decision making."

Students are gradually introduced to increasing levels of responsibility for patient care. The apprenticeship is usually conducted in an environment where there is an appropriate mix of medical challenges and educational supervision. Little emphasis is usually given to formal instruction in the elements of the decision process, and the student is expected to acquire clinical judgment by learning to imitate the patterns of his or her instructors.

This method of teaching has several potential weaknesses: (1) each student's educational experience is relatively unstructured and may be erratic; (2) skilled lecturers with adequate time for close student supervision may not be available; and (3) a variety of "patient material" is required for each student to learn about various disease entities. As the student advances in his or her training, the degree of supervision is reduced; there is less monitoring of performance, less feedback, and less immediate correction of error. As a result, if a student does develop inadequate behaviour patterns, he or she will tend to continue to use them because he or she does not know any better.

Influences such as the early introduction of clinical responsibility, class size, political and economical factors, etc. have placed an extra load on the teaching faculty. This in turn has resulted in far less opportunity for individualized student interaction and supervision. Because the patient population in many of the teaching hospitals is becoming so specialized, a medical student may encounter only a few patients with common diseases for whom he or she can have personal responsibility, and on whom he or she can exercise his or her clinical judgment and practice his or her skills.

The emphasis on hospital-based teaching has resulted in shorter time being allocated to the General Practice curriculum. An appropriate example is the undergraduate General Practice curriculum in Sheffield which is allocated 5 weeks in 5 years. During this period, the student is assigned to clinical training at designated surgeries with one day a week for lectures and tutorials.

Overall, undergraduate students of General Practice in this country are given less time to improve and extend their knowledge in patient management and case diagnosis. This may have negative consequences on young practitioners starting work in a community. The Royal College of General Practitioners have put forward five areas in which a practitioner must acquire and learn to integrate knowledge, skills and attitudes in his or her work. These areas are:

1. *Clinical Practice (Health and Disease):* This includes details of the natural history and management of diseases.

2. *Clinical Practice (Human Development):* This involves knowledge of the range of normality and the changes that occur in normal human development.

3. *Clinical Practice (Human Behaviour):* This involves knowledge of normal human behaviour.

4. *Medicine and Society:* This include knowledge of individuals as members of a larger group and the knowledge and expectations of the society to be served.

5. *Medical Facilities:* The tools for the job. These include the general consultation model, hospital and other services, medical records, equipment, and staff.

Given the comprehensive obligations to patients and community, would-be General Practitioners should be better prepared than they are at the moment.

2.2 COMPUTER-BASED EDUCATION IN MEDICINE

The use of the computer in medical education has been in evolutionary development since the early 1960s. Its adoption, however, has been less widespread than it should be. Computer-Assisted Learning (CAL), a term which includes basic computer-assisted instruction (CAI) and intelligent computer-assisted instruction (ICAI), enhances learning, allowing the student the discretion of content, time, place, and pace of instruction. The information communicated by these systems can take several forms, some better suited to undergraduate medical education, others more applicable to graduate and continuing education.

The use of CAL systems in undergraduate medical education in the UK is still far from clear. The main problem is the lack of funding for the research into and development of such systems. The majority of medical teaching systems were developed as part of research projects. Usually, once a project has terminated the system became obsolete.

The main problem is the unrealized value of CAL systems in medical education. As outlined earlier, at some stage, a student has to begin taking responsibility for his or her decisions when managing patients. Although such responsibility is gradually introduced and increased in levels during clinical training, an important ethical point emerges. Using a CAL system for student training can be justified on the following points:

1. An extensive mix of "patient material" can be presented for the student to assess without being dependent on the availability of suitable cases in the clinics or wards.

2. Patients are not disturbed or put under any risk while students are learning.

3. Individual experience is acquired. Usually, students work in groups during their clinical training in wards or clinics. Computer-based education systems provide a one-to-one correspondence between the student and the machine, hence facilitating individualization in learning.

4. Cases can be followed from beginning to end at the student's own pace without any time constraints or other factors.

5. The development of standardized testing procedures for measuring students' acquisition of problem-solving skills. Such procedures would be of considerable value to the teaching faculty in student assessment.

A recent survey conducted by the University of Dundee Medical School [Florey, 1988] to describe the state of CAL in British medical schools has come up with a number of findings which confirms the points stressed in this thesis. These can be summarized as follows:

1. All the CAL systems used in the medical schools were developed using the BASIC programming language and the MICROTEXT authoring language. Neither of these languages is ideal for such applications. The main issue, here, is that the majority of these systems were developed by the medical doctors who are usually inexperienced in high level computer languages. Collaboration between medical schools and computer science departments is very rare.

2. The costs involved (hardware, software, personnel, etc.) in the development of an educationally satisfactory CAL systems represent a heavy investment to most medical schools.

3. The lack of any coordination between medical schools involved in the development of CAL systems. Most medical schools try to develop systems for the purpose of competing with other schools rather than trying to combine and share resources and ideas.

The overall viewpoint is that sophisticated computer-based medical education systems will not be seen in use at the medical schools in this country for some considerable time.

2.3 THE UNIVERSITY OF SHEFFIELD INITIATIVE

At the Department of General Practice, University of Sheffield, a commitment has been made to introduce a CAL system in the next General Practice curriculum starting in April 1990. An expanded version of the system developed in this joint project will be used for this purpose. A CAL laboratory based on several IBM PC compatibles linked in a local area network will be setup for use by undergraduate students during the course of their studies.

The objectives of the Department can be summarized as follows:

1. To increase the clinical content of the undergraduate course in Primary Care.
2. Avoid the overload of the current teaching system (in particular human General Practice tutors).
3. Extend the availability of tutorial services.
4. Improve over the current methods used in the assessment of student performance.

The overall objective, however, is to provide the basis for CAL systems to be introduced in General Practice departments throughout the country. It is hoped that the lessons learned from developing and implementing this system will be taken into consideration when developing future systems for the Primary Care environment.

2.4 TEACHING STRATEGIES USED BY COMPUTER-BASED MEDICAL INSTRUCTION SYSTEMS

Among the different types of teaching or learning strategies, the following four are the most common in medical Computer-Assisted Instruction and Intelligent Tutoring Systems:

1. Drill and Practice
2. Discovery Learning
3. Self Assessment
4. Conjectural Teaching

Other well known strategies such as Cognitive Apprenticeship, Successive Refinement and others, are not appropriate to medical education.

The sections that follow, describe in some detail these strategies or approaches.

2.4.1 Drill and Practice

Systems implemented on the basis of this teaching strategy are based on the original programmed learning concept of the behaviourist psychologist Skinner. Briefly, the idea is that learning takes place by the reinforcements of success. A correct response should always be followed by positive reinforcement. Incorrect responses are ignored. Since learning is held to take place only when a correct response is reinforced, learning material must be prepared in such a way that correct responses are most likely to follow. This means in practice, that material must be split into small sections, each of which represents such a small advancement in understanding that success is guarantied and rewarded before moving on to the next section [Clayden *et al.*, 1988]. Skinner emphasized that once mastery is reached students must be dissociated from this approach in order to avoid rapid weakening of the response. To do this, he recommended shifting continuous reinforcement to a pattern of intermittent reinforcement [Skinner, 1968].

This theory was tested on animals with impressive results. These tests led to the development of teaching machines and the introduction of programmed learning. However, these machines were not suitable due to the cost and difficulty of implementing and using them. As a result, computers seemed the perfect answer. The early days of CAL saw an overwhelming use of the drill-and-practice type of programmed learning.

However, the use of this type of system revealed a number of shortcomings as Clayden and Wilson [Clayden *et al.*, 1988] point out:

> "...many teachers, particularly in the UK, were already sceptical about the value of programmed learning, and their misgivings were not confined to the technology of the concept. They were aware that pupils soon became weary with the repetitive nature of the learn, test, reward sequence, and that the small step-size which guarantied success also guarantied boredom. Moreover, many of them questioned the whole basis of behaviourist psychology, and felt from their own experiences that human beings learn as much from mistakes as from success, providing that those mistakes take place within an environment where it is possible to discover the reasons for the errors, and to propose alternative methods of overcoming them. For these reasons, many teachers were less than enthusiastic about the advent of the micro in the classroom regarding it as simply the latest of a series of educational gadgets based on a wholly discredited fashion of teaching."

On the basis of these reasons, this particular style of teaching was not favoured in this country. A modified programmed learning strategy is currently in use in some of the available packages whereby the system allows the learner to proceed at a much greater rate, while still informing him or her of any errors and misconceptions.

In terms of CAL in general, this type of system takes the form of branching programs, testing the student's knowledge, and allowing remedial routines to be entered if a certain pre-determined score has not been achieved. Basically, these systems are electronic page turners.

They introduced sophisticated graphics, and used a variety of mechanisms (e.g. light pens, mice, VideoDisks). Yet, they remained somewhat complex drill and practice systems unable to recognize anything beyond the expected response. These systems represent the bottom line of CAL, which include all pre-AI instructional systems, since they give no freedom of choice to the students, but guide them through the pre-prepared dialogue of the teacher. These systems are relatively easy and cheap to prepare, and examples of this type of program can be bought for many preclinical subjects, e.g. the eye, the heart, etc. [Clayden *et al.*, 1988]

Another class of systems that fall within this category are those based on Multiple Choice Questions (MCQs). MCQs form a large part of a medical student's evaluation procedures. The inclusion of MCQ data banks, with test items categorized according to content and level of difficulty, in a CAL system is probably well justified. The use of MCQs as a form of teaching is discussed further in the next chapter.

2.4.2 Discovery Learning

This type of teaching owes much to cognitive psychology. This technique is much favoured in the English educational establishment, and utilized in the familiar 'project' method of learning.

In terms of CAL in general, greater emphasis is placed on the learners and their needs, and much less on the instructional material, i.e. the program. Here, instead of the program being in total control, with the learners as passive responders, the program allows and encourages them to choose paths through the knowledge base, to explore what is hidden and what is gradually revealed. Thus the program acts as mediator between the subject domain and the learner. Typically, the program sets up a model of reality for the learner to observe the behaviour of the presented situation or to practice problem-solving and decision-making [Clayden *et al.*, 1988].

This model of reality often takes the form of a simulation. A simulation is appropriate when reality itself is too complex, too dangerous, too expensive or otherwise inappropriate for the learner. In medicine, simulations are indispensable since these would-be doctors will, at some time in their future, be dealing with the health of fellow human beings.

In medical education, a simulation usually takes the form of a case study. Two types of simulation exist: Static and Dynamic. Static simulations typically involve diagnosing and possibly treating a fixed patient problem. The learner's aim is to elicit the information necessary for diagnosing and managing the particular case. Dynamic simulations, on the other hand, employ an underlying, dynamic model of the patient. In addition to eliciting diagnostic information, the learner can observe changes in the patient over time and can make therapeutic decisions and observe the results.

Patient management simulations of this kind are very time-consuming to construct where sufficient historical details must be included to add realism and human interest to the problem. Furthermore, dynamic simulations require the use of mathematical and statistical models when simulating the results of the learners actions.

Both static and dynamic simulations have proved popular with students. Furthermore, the addition of visual material such as colour slides stored on a CD-ROM or video sequences on videotape greatly increases the interest and involvement. However, this also increases the capital cost of equipment, and increased sophistication means more complex programs, and therefore the greater need for professional help.

Computer simulations cannot hope to model every facet of the real-life situation, and it is all too easy to present a plausible representation of reality. It is obviously important that the program makes this perfectly clear to the learners, and refers them to other sources of information and data. Nevertheless, this type of teaching would seem to offer a great deal in medical education, providing that teams of authors and experts can find the time to put together a realistic and worthwhile model of their subject area.

Patient management simulations are discussed further with full examples in the following chapter.

2.4.3 Self-Assessment

In the previous sections we concentrated on the process of teaching. However, CAL is also concerned with learning. This is significant since the term CAL places a greater emphasis on the learning rather than the teaching aspect of education. Learning may well involve other activities such as the preparation or ordering of notes, writing of essays, gathering and analysis of results and storing and retrieval of information. In all these activities, the computer has revolutionized students' ability to process rapidly large quantities of information. Tasks which were previously time consuming and error prone can now be handled quickly and with confidence. In self-assessment systems, the student is given a complete control of his or her learning situation. At present, a number of systems have been built trying to encapsulate this particular form of learning. One example of a self-assessment system that allows the student some freedom is ENMESH (see section 3.4 for full details on this system).

As well as this entirely content-free aspect of CAL, this style embraces an exciting form of learning which has yet to be developed fully in this country. Systems developed on the basis of this style of learning are identified as *Learning Environments* [Clayden *et al.*, 1988].

In a learning environment, students learn what they want, when they want it, and at their own pace. However, there is a price to pay for such a system. The software is very difficult to write particularly in field such as medicine, and an effective system requires the dedicated collaboration of program designers, educationalists and subject specialists over a considerable period of time. Current medical learning environments are not widely available and most are based on expensive research workstations. Learning environments in fields such as mathematics, programming languages (e.g. LOGO) are more widely available than their medical counterparts.

2.4.4 Conjectural Teaching

This is an extension of the Discovery Learning technique. It differs from the former, however, in that it allows the learners not merely to explore the simulated model, but actually to change the nature of the model, or at least the parameters underlying it. It is also described as the *'What if...?'* approach to learning [Clayden *et al.*, 1988].

Conjectural teaching represents the dynamic type of simulations in Discovery Learning in the sense that the learner can change the system parameters and observes and draw conclusions from the results provided by the system. Systems built around this paradigm tend to be mathematically-based where any adjustments to the parameters of the model can be accomodated and explored. Examples of systems based on this paradigm include McPuff and McPhee, which are built on the Apple Macintosh microcomputer [Clayden *et al.*, 1988].

This approach is most suited to expert systems which have the ability to explain their rules. At present, the use of expert systems on microcomputers is still in its infancy. A number of expert system shells are now available for use on some microcomputers. However, these are very expensive and represent a major investment to their users. In addition, some are not suitable for handling deep knowledge as required in the medical domain.

2.5 CURRENT TEACHING METHODS

The teaching and assessment of medical students whether undergraduate or postgraduate during their clinical training and their formal teaching involves the following four methods:

2.5.1 Case Histories

In this method, the student is provided with a patient case history and asked to collect further data from the patient for the purpose of diagnosis and management. A form of simulated case histories called Modified Essay Questions (MEQs) is used in the assessment and examination of General Practice students. This is a cost effective method of patient simulation since the simulation is printed on paper. It is also flexible as opposed to computer-based systems which require the setting up of special CAL laboratories.

Case histories are the most widely used method of teaching in computer-based learning systems. Systems such as the Glasgow system [Murray, 1977], CASES [Verbeek, 1987], and MEDICS [Hagglund *et al.*, 1981; Wigertz *et al.*, 1983] are all based on the case history presentation method (see section 3.5.1 for brief descriptions of these systems).

2.5.2 Multiple Choice Questions (MCQs)

As pointed out earlier, MCQs represent a very important method of student assessment and evaluation. MCQs range from the simple *'true-false'* type through *'five choice completion'*, *'five choice association'*, *'excluded term'* and *'relationship analysis'* [Marmion *et al.*, 1971]. An MCQ usually consists of true and false facts. The ratio of true to false facts depends on the relative easiness of the tested facts (i.e. difficult questions have a higher number of true facts, and vice versa). MCQs are commonly used in undergraduate and postgraduate examinations.

For many years, computers have been used to assist the marking of the MCQs in examinations. The candidates are required to record their answers on special sheets that are suitable for optical scanning. These sheets are then 'read' by commercial scanners and the data fed directly into the marking computer.

CAL systems based wholly on MCQs represent an under utilization of the power of the computer even though the computer can point out:

a. Poorly answered questions.

b. How marks from various questions compare with previous tests.

c. Which questions gave good discrimination, i.e. good students answered them correctly and poor students incorrectly.

In addition to wasting computer resources on such simple systems, they are in a way unsuitable for teaching purposes mainly because they prompt and provide hints to their users. This is not particularly useful from an educational point if the student is a confused one.

Software that is based on MCQs requires an authoring system for the expert author to "hand-feed" the questions and their relevant parts. Authoring languages are currently available on the market offering this type of question generation. They also have the ability to generate the simple 'true-false' questions. The main problems with such languages are their high cost and their inability to be tailored to the end-user requirements. Other problems with authoring languages are listed in Chapter 5 table 5.1.

Multiple choice questioning is sometimes used as part of Case History-based systems. GUIDON (see section 3.5.2 for a brief description), for example, generates its own MCQs from previously known facts. *P.C.Tutor* uses a similar approach to that of GUIDON in generating some MCQs. This approach is discussed in Chapter 5.

2.5.3 Data Interpretation

In Data Interpretation, the student is provided with some information about the patient which includes the results of laboratory tests made on the patient. The student is asked to interpret such results for the purpose of diagnosing and suggesting therapy. The types of data used in this method include all laboratory test results (numeric or otherwise). No visual data is used in this method.

GUIDON [Clancey, 1987a] uses this method as part of its tutoring strategy. This method is also used in *P.C.Tutor* (see Chapter 5 for full discussion).

2.5.4 Visual Media

Visual Media include X-Rays, Ultrasound, ECGs, CT-scans, Photographs of injuries, rashes, and so on. In this method, the student is given limited information about the patient and supplied with a complete set of visual data on the particular patient for him or her to interpret. A combination of data interpretation and visual media is always used during student assessment and examinations. For example, supposing that the presenting symptoms and clinical signs of a patient points towards Adult Pulmonary Tuberculosis. Given the results of radiology (X-Rays) and bacteriology tests, a student should be able to confirm this infective disease [Hodgkin, 1985].

This method requires the use of expensive hardware when used as part of a computer-based teaching system. CASES [Verbeek, 1987] is one of the documented systems that uses VideoDisk. The main difficulty with VideoDisks lies in the gathering of the necessary material, its preparation (i.e. the pre-mastering stage), and its storage on disks.

Using visual media in the form of Interactive Video has increased during the last few years in several fields of medicine (none of these were aimed for use in the General Practice environment). One of the well known VideoDisk-based systems is the St. Bartholomew's Hospital system [Ingram *et al.*, 1989] designed to assist the teaching of three medical topics: (1) Intensive Care, (2) Diagnostic Imaging, and (3) Accident and Emergency.

The use of visual media with *P.C.Tutor* was considered by the research team and then had to be abandoned due to lack of funds.

CHAPTER III

REVIEW OF MEDICAL TUTORING SYSTEMS

3.1 INTRODUCTION

Medical teaching systems have always been based on the process of diagnosis. In fact, diagnosis in all domains, plays a special role in instructional programs since it is a common problem-solving task that researchers and educationalists attempt to teach, e.g. debugging computer programs, fault finding in electronic circuits or equipment, and disease diagnosis. More significantly, as Clancey [1988] points out, diagnosis plays a central role in teaching itself. He states that:

> "In instruction, we monitor the behaviour of the student (a cognitive system), look for discrepancies from the ideal specification (target problem-solving model), track discrepancies back to faults in the student's presumed world model or inference procedure, and 'repair' the student by instruction. This process of causally tracking backwards from discrepant reasoning behaviour to hidden faults in a cognitive system is called *diagnostic modelling*."

The computer and medical literature of the last twenty years has described several projects ranging from the use of computer technology in medical education and training to its use in assisting clinicians in their fields. The educational projects used a variety of instructional techniques that have been developed by cognitive and computer scientists. These projects have ranged from simple drill-and-practice systems to sophisticated dynamic simulations of body mechanisms and clinical treatment. Evaluations of these systems, as reported by the systems developers, have always resulted in favourable responses from users. However, few of the systems have been used far beyond their places of origin and few have survived long in use [Ingram, 1988].

This situation can be attributed to a number of factors. First, building this type of system requires the expertise of a medical doctor who has considerable time on hand to be able to work on such a project. Finding this person is difficult. In addition, medicine is a specialized domain limited to certain practitioners as opposed to mathematics, computer programming languages, etc. where, for example, a computer expert can develop a program for teaching mathematics. Second, some doctors argue that a computer-based teaching system would not have any significant effect on the learning of a medical student. This argument is usually voiced by those who either feel threatened by computer technology or afraid of using this technology. Opposition of this nature is becoming less common due to the greater impact of information technology on education and the daily work of doctors. One would argue that computer-based teaching systems will never attempt to replace clinical training. In fact, their purpose is to complement such training and provide a method for clinical knowledge reinforcement. A third factor which is attributed by medical personnel, is the fast rate of technological advance that has resulted in some of the systems becoming obsolete. A fourth factor is the changing perceptions of computer users as to what constitutes an acceptable system.

The main reasons for failure of some of the medical systems summarized by Ingram [1988] and based on an editorial review published in 1977 [Computers and Biomedical Research, No. 10, pp 199-204, 1977] can be outlined as follows:

a. Poor facilities for interaction with the user. The system did not exceed the user's own capabilities and therefore had only novelty value.

b. There was no direct impact on patient care.

c. The system was poorly designed from the point of view of maintenance and portability. As a result, subsequent change and updating of the system became too difficult.

d. The designers had failed to learn from their own and other people's mistakes.

Such criticisms reflect unclear or inappropriate objectives, inappropriate or poorly managed technology and partly simply human failing [Ingram, 1988].

One area of medicine where such factors played a major role is the Primary Care environment. Only two systems have been developed for this environment. The majority of projects were based on specialized fields of medicine, e.g. pathology, anaesthesia, cardio-vascular system, etc.

In this chapter, I will be examining the main differences between Primary Care systems and hospital-based systems. I will also be evaluating some of the documented medical tutoring systems developed from mid- 1970s to the current date.

3.2 WHAT IS A PATIENT MANAGEMENT PROBLEM (PMP)?

The underlying point in discussing PMPs is the fact that they represent the basis of a large number of medical CAL systems.

PMPs are defined by Harden [1983] as:

"...exercises which simulate the decisions a doctor has to make in the diagnosis and treatment of a patient's illness."

There are three types of PMPs: *Linear Structured*, *Branching Structured* and *Open*. The majority of systems discussed in this chapter are of the type Branching Structured. In this type the student goes through a pre-defined route of required decisions. Each decision represents the parent node of a tree. The student has to select the appropriate decision from a set of decisions at each stage of the route until the end of the case. Feedback is provided to the student on his or her selected decisions based on a consensus of a group of experts who act as a criterion reference group for the particular problem. By simulating a real situation, PMPs allow a student to practise, assess or be evaluated on an aspect of clinical competence, without the use of patients. PMPs are also valuable for individual learning and for small group teaching. They allow the student to follow a case through to the end and to get the feeling of continuity which is often lacking in short-term teaching in the clinics and wards. The type of cases that are suitable for use in PMPs are so-called 'cold' cases, where the patient is being assessed over a reasonable lengthy period of time, and where there is no short term urgency in the management of the case [Taylor, 1975].

The main weakness of the Structured type PMPs is that the student is strictly guided by the PMP design to the objective set by the expert teacher.

Linear structured is similar to Branching Structured except each decision represents a point on the pre-defined route.

An Open PMP differs from the other two by allowing the student to follow any route by moving back and forth between available decision actions. This type of PMP is suitable for simulated learning environments where the student can learn from trial and error. The tutoring system developed during this research is designed on the basis of the Branching Structured and Open PMP. It is not, as yet, a full PMP mainly because the patient does not change over time.

3.3 THE DIFFERENCES BETWEEN PRIMARY CARE TEACHING SYSTEMS AND HOSPITAL-BASED SYSTEMS

The differences outlined in this section apply to health care in the United Kingdom and any countries with similar care (e.g. Sweden). The aim of all clinicians (General Practitioners and hospital doctors) is to cure or treat and bring the patient back to health as far as possible. The aim of a General Practitioner (GP) faced with a patient problem is to treat it within the shortest possible time. In order to do this, the GP must have full knowledge of the diseases that are common in the particular community. This knowledge is always cross referenced during a consultation and the appropriate testable hypothesis or differential diagnosis is formulated. The differential diagnosis consists of a list of plausible diseases that have common presenting complaints. For example, if the main disease is Rubella (German Measles), several diseases may be confused with it. These diseases include Measles, Scarlet Fever, Infectious Mononucleosis (Glandular Fever), and some allergic rashes [Hodgkin, 1985]. Given information obtained from the patient and/or observed by the GP during the consultation, the GP refines (or streamlines) his or her differential diagnosis to a few diseases or problems. On the basis of the streamlined differential, the GP formulates an appropriate management plan for treating the particular case. The important issue, here, is that a GP does not require a definite diagnosis for the purpose of managing a patient problem. Furthermore, GPs do not have the immediate availability of laboratory tests and other facilities that are available in hospitals for making a final diagnosis. A GP sometimes formulates a management plan on the basis of a limited set of evidence (such evidence includes vague, unstructured and partly irrelevant information given by the patient, and information collected by the GP through questions and observation) and on his or her pattern matching skills and experience. The two important tools for a GP managing a patient problem are time and the recurring patient encounter. This is not to say that disease diagnosis is not important in the General Practice environment. In fact, a GP should be able to formulate, and follow-up an appropriate differential diagnosis with each patient problem. He or she must consider all alternatives before taking up and communicating his or her actions to the patient. Overall, a GP processes more information than a hospital doctor.

Hospital doctors, on the other hand, have at their disposal all the facilities for making definite diagnosis prior to treating the patient. In addition, hospital doctors confronted with a patient problem always follow a fixed sequential route for information gathering prior to diagnosis and treatment. Usually, once the information gathering process is completed, a doctor starts by generating his or her differential diagnosis which is further tested and refined accordingly prior to treatment. The procedure from collecting the information to the actual treatment is not applicable to General Practitioners. When a GP is faced with a patient problem, he or she must eliminate any irrelevant information and generate an appropriate differential diagnosis. The processes taken by the GP are executed in parallel rather than in sequence. Because a GP is working in a larger domain than that of a hospital doctor, he or she must have a comprehensive knowledge of all the diseases and be able to apply his or her clinical judgment and skills in the short time of the consultation.

The majority of known teaching systems are based on the hospital work environment. This is, probably, due to the vast medical field that applies to the General Practice environment.

To summarize the important medical point of this research, the open strategy of diagnosis adopted by all GPs, where there is no need for an immediate diagnosis at the first patient encounter, represents the overall difference between the two environments. Therefore one needs to stress that, in General Practice, the main emphasis is on all areas of patient management (including the social and psychological aspects) as opposed to disease diagnosis in hospital work. Students using *P.C.Tutor* are made aware of the differences in content and approach between the care of the patient in hospitals and in General Practice.

3.4 PRIMARY CARE SYSTEMS

Primary Care teaching systems developed in General Practice departments are very rare. The few systems that are available were developed by pharmaceutical companies promoting their products. These commercial systems are not designed for use as part of the Primary Care curriculum in medical schools. They are intended to "teach" practising GPs to use their products. Such systems are based on simple multiple choice questions.

The only documented system that claims to be aimed at General Practice and which was developed by a General Practice department is the Glasgow University system [Murray *et* *al.*, 1976; Murray, 1977]. Written in FORTRAN IV, this system consists of three models; (1) Case History, (2) Emergency Model, and (3) The Three-Disease System. The Case History model is a simple Structured Branching PMP. Expert prepared cases are stored in a database for the system to select from. For each case students are provided with a folder containing the patient's previous medical history and details of family and occupation. Actions to be taken by the student are displayed in menus for him or her to rank according to their likelihood. Once the student has ranked these actions the system displays the expert's opinion in comparison with the student's choices.

At the end of the session, the program displays its recommendations for the case and thereafter displays a summary of the incidence and prevalence of the condition in the community.

Figure 3.1 shows an example from the case history model. Actions or decisions are ranked or valued between 1 and 5, with 5 representing the most important action that must be done always, and 1 representing an action that must never be done:

5 - must do
4 - would do
3 - could do
2 - should not do
1 - must not do

where 5 and 1 must always be correct and an error of ± 1 on 2, 3 and 4 was allowed.

Note that the figures in italics represent the experts' decision values which is followed by the students chosen values in brackets.

Mr Ross is 42 years of age. He is a steel erector, smokes 30 cigarettes a day, and is a heavy drinker. Mrs. Ross is 40 years old and has a part-time job as a home-help.

{Other details about the family are given to the students plus information about the simulated surgery. Past patient medical history and family details are given to the students in an A4 folder}

....

Mr. Ross consults you one evening about a rash which you diagnose as pityriasis rosea.

You explain this condition to him, and as you have not seen him for three years you decide to check his blood pressure - this you find to be 185/125.

Do you:

1. Carry out a full examination	*3* (5)
2. Refer him to a hypertension clinic	*2* (3)
3. Reassure him and say elevation probably due to excitement	*4* (2)
4. Admit him to hospital	*1* (2)
5. Arrange to review in one week	*5* (5)
6. Lie him down for five minutes and take blood pressure again	*4* (5)

FEEDBACK

1. No point in full examination as no diagnosis can be made until adequate baseline readings are made.

2. Hypertension clinic not yet indicated - baseline readings and first line investigations required first.

3. Reassuring him would be reasonable action as one reading is often unreliable - but at this level likely to be significant.

4. Admitting him to hospital totally wrong at this stage - acute hospital beds are required for acute emergencies.

5. Arranging to review in one week is the treatment of choice - most experts agree that three baseline readings are required.

6. Repeating after five minutes is reasonable action but significant change unlikely. More definite follow-up required.

{When the options are ranked, the program continues by presenting further details on the patient and his problem development}

Figure 3.1 - Excerpt from the Case-History Model of the Glasgow University System [Murray,1977]

The second model, the emergency model, is designed to allow students to explore strategies of investigation, diagnosis and treatment while working against the clock. Such a time-dependent model is of no use to GPs since all emergencies are sent to hospitals. In fact, the Glasgow system is being used at a Sheffield hospital for the occasional training of medical students. The third model is a simple differential diagnosis generation model.

Two serious problems with the Case-history model are the prompting and system guidance of the student throughout the session. Such total constraint imposed upon the student is not favoured by a large number of ITSs' researchers [e.g. Elsom-Cook, 1988]. The Case-history model is nothing more than a simple electronic exercise book.

To conclude this review of the Glasgow system, I must point out that the importance of patient management was concluded by Murray [1977] from the trials of the system even though the system was not built according to the general consultation model followed by General Practitioners (see section 3.7 for details of this model).

Other American-based systems briefly documented in some of the medical journals (e.g. the Computerized Clinical Patient Problem C2P2 [Pickell *et al.*, 1986]) on family medicine which is somewhat similar to General Practice in the U.K. follow the same problem solving methods used in hospital-based systems.

An undocumented system, called ENMESH developed at the Department of General Practice, University of Liverpool, was demonstrated at the *Computers in Medical Education* conference, Cardiff, December 1988, and at the AUTGP conference, Liverpool, July 14th, 1989. This system, originally written for use on a BBC B (TM) microcomputer running the old CP/M(TM) operating system, and converted to run on the IBM PC used simple pre-entered case histories and free-text sentences for user interaction. ENMESH is a self-assessment-type system (i.e. the student assesses his or her work by comparing his or her answers with those supplied by an expert). The system's scenario starts as follows:

At the start of the first session the student is given four case histories to manage in order to find out his or her areas of weakness (Physical, Psychological, or Social). The way these cases are presented is the same in all subsequent sessions.

During a session, cases are first displayed on the screen and the student is required to enter his or her appropriate actions that are relevant to the case requirements. Student actions are entered as free text with no validation whatsoever. Once all actions are entered, the system thereafter displays a list of alternative actions stated by expert GPs. The student is then asked to confirm his or her actions by comparing them with the expert's actions. The system goes through a number of cases in this linear manner before giving the student his or her score on the three above named main areas or categories in the system.

The only controls that prevent a student from cheating is the storage of all student actions on file for the teacher to examine. Figure 3.2 shows an example of a Physical case.

SECTION - PHYSICAL.

Mr W aged 51 works hard in the Careers Service. He appears happily married and has 2 sons who are away at college. A few months ago he developed a left inguinal hernia; ten weeks ago this was repaired by a local surgeon. Post-operatively he developed a wound abscess which discharge profusely but is now drying up.

When you saw him 3 weeks ago he felt tired but you thought he would soon be ready for work. Today he says that he is keen to return to work but remains very tired. He is sleeping well and gaining weight.

List up to 6 causes of his tiredness which you would consider.

{The student here enters up to six causes}

GP RESPONSES (1-10).

1) Complications of surgery eg anaemia/persistent infection
2) Physical illness unrelated to the hernia or its treatment
3) Psychiatric illness
4) Is needed at home because of family/relationship problems
5) Fears about capacity to cope physically or emotionally at work
6) Conditions precipitated by surgical trauma eg UTI/renal failure
7) Protracted physiological post-operative debility
8) Drug/alcohol dependency
9) Side effects of anaesthesia/drugs eg sub-acute liver damage
10) His perceptions of the nature of the illness/complications

11) Valid but not listed
12) Inappropriate

STUDENT RESPONSE.

{All student responses are listed one at a time for him or her to enter the number from the above list that corresponds to the particular response}

....

{After seeing a number of cases, the system presents the student with his or her score}

YOUR SCORE ON THE THREE CATEGORIES OF RESPONSES IS AS FOLLOWS:

PHYSICAL

0	1	2	3	4	5	6	7	8	9	10
:	:	:	:	:	:	:	:	:	:	:

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX----------------
(unsatisfactory)      (borderline)      (satisfactory)      (exceptional)

PHYSIOLOGICAL
0    1    2    3    4    5    6    7    8    9    10
:    :    :    :    :    :    :    :    :    :    :
XXXXXXXXXXXXXXXXXXXXXXX---------------------------
(unsatisfactory)      (borderline)      (satisfactory)      (exceptional)

SOCIAL
0    1    2    3    4    5    6    7    8    9    10
:    :    :    :    :    :    :    :    :    :    :
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX----------
(unsatisfactory)      (borderline)      (satisfactory)      (exceptional)
```

Figure 3.2 - An example from the ENMESH system

ENMESH is not a PMP or a simulation system of any kind. It is a very simple linear system. The teaching strategy adopted by this system is similar to any simple written exercises used in all schools and universities. The system is a simple electronic exercise-book. It is a known fact that students interest in this kind of system diminishes after a few cases. In addition, the student is confined to the information given in the case text. He or she cannot ask for additional symptoms or make any examinations. Furthermore, due to the static nature of the cases, a student can memorise any GP responses for use when the same case is selected for him or her. To overcome such problems, the system has to have a very large bank of cases for varied presentation. It also requires a way for keeping track of all cases presented to each student to avoid constant repetition of cases.

Overall, the ENMESH system offers nothing towards the development of medical CAL systems, let alone the ideal system.

Finally, a somewhat different teaching system being developed in a number of universities for use in the General Practice environment is the type aimed at patients' education. Such systems provide patients with printed advice sheets describing the nature and management of their problems, detailed information about their prescriptions, and tests their general knowledge in the particular domain. Systems of this kind are already being used at some General Practice surgeries and private clinics, and dental surgeries in the USA, and Europe [Fos, 1988; Guardian Newspaper, 1989].

3.5 HOSPITAL-BASED SYSTEMS

A large number of hospital-based systems covering a wide range of specific areas of medicine have been developed over the last twenty years or so. The relevant systems briefly discussed in this section are well known in the medical and computer science fields. Medical teaching/tutoring systems can be categorized into two computer science related categories: (1) Computer-Assisted Instruction (CAI) systems, and (2) Intelligent Computer-Assisted Instruction (ICAI) (or as known by some authors as Intelligent Tutoring Systems (ITSs) [Sleeman *et al.*, 1982]).

CAI systems are simple electronic "page-turners" or simple practice monitors, that prints prepared problems and responds to students solutions using pre-stored answers and remedial comments. ITSs, on the other hand, have their subject material represented independently of the teaching procedures, thus problems and remedial comments are generated differently for each student.

To be more specific, ITSs differ from CAI systems in that they distinguish among the components of the tutoring system itself and use heuristics to refine the decisions made in each component. The main components of an ITS are:

1. Expert Module
2. Student Model
3. Tutoring Module
4. Communications Module

Each of the components later became the focus of research in its own right (see section 3.5.2 for details of these components). A variety of ITSs have been built as research projects in several domains. Examples of some of the known systems include:

1. Mathematics - e.g. Kimball's [1982] self-improving tutor for symbolic integration, and O'Shea's [1982] quadratic tutor.

2. Mathematical-based Gaming Environments - e.g. WEST [Burton *et al.*, 1982] the coaching system designed to teach the appropriate manipulation of arithmetic expression in a computer game environment, and the WUMPUS system [Goldstein, 1982].

3. Programming Languages - e.g. PROUST [Johnson *et al.*, 1987] the knowledge-based system that identifies bugs in Pascal programs and suggests remedial actions.

4. Medical Diagnosis - e.g. the GUIDON system [Clancey, 1987a] (see section 3.5.2. for an overview of this system).

5. Electronic Circuit Troubleshooting - e.g the SOPHIE systems [Brown *et al.*, 1982].

6. Geography - e.g. the South American geography tutor SCHOLAR [Carbonell, 1970].

With the exception of GUIDON, I will not attempt to review or discuss any of the above ITSs for the basic fact that they are irrelevant to this research.

3.5.1 Computer-Assisted Instruction Systems

CAI systems are the most common types of instructional programs developed in medical schools throughout the world. The systems discussed in this section are built on a combination of three teaching approaches: the *Drill-and-Practice*, *Discovery Learning*, and *Self-assessment* approaches. The systems discussed in this section were chosen for two reasons:

1. They are well known documented systems; and

2. The combination of these systems cover a wide range of instructional features.

3.5.1.1 The CASE System

The first example of CAI systems is the early CASE system [Harless *et al.*, 1971]. In this system, the student goes through various stages of the Secondary Care model, i.e. he or she starts by obtaining the patient's medical history, symptoms, and performing any appropriate examinations and laboratory tests. Once all the facts have been gathered from this "encounter" (as it is called by Harless) the student enters his or her diagnosis and suggest treatment and therapy. The student is scored by the system, using scoring criteria set up by the person writing the cases. The scoring system is based on how well the student has obtained the relevant set of information and his or her diagnosis and treatment of the case. The CASE system used Natural Language Processing (NLP) in its enounter. Figure 3.3 shows an excerpt from a CASE tutorial.

{Having described the work environment and introduced the patient details, the program continues as follows:}

Student : What brings you to see me?

Computer: Well, I'm just tired all the time. I don't ever feel like I'm rested and all my energy is gone.

Student : How long has this been going on?

Computer: About six months ago I started feeling tired all the time. I finally felt like I had to do something about it.

Student : What did you do?

Computer: Are you still talking about the my being tired?

Student : Yes.

Computer: Well, I've been taking vitamins, but they don't seem to help. Maybe you can give me something stronger.

....

Figure 3.3 - Excerpt from a CASE tutorial [Harless et al., 1971]

The use of NLP was the main factor in the failing of this system. The system had limited vocabulary and most of its processing time was dedicated to NLP.

3.5.1.2 The THYROID Program

The THYROID program [Steele *et al.*, 1978] is a simple CAI system which teaches problem recognition and management of 20 thyroid disorders. The system uses the case-history simulation method of teaching. Medical knowledge including all actions to be taken by the student is stored in tables with associated indexes. The actions are known as items. These tables consist of:

- History Taking
- Physical Examinations
- Laboratory Tests
- Therapeutic Options
- Diagnosis

As an example, in the History Taking table, the following is one of the questions in this table:

106 HAS YOUR WEIGHT CHANGED MUCH RECENTLY?

For each of the 20 diagnoses, an estimated probability of a given response being given to a particular history taking question or physical examination action, is entered in the database. With each question in the database a set of responses are associated with it. The computer selects one of these responses on the likelihood of a given response occurring for a specific thyroid disease.

During a tutorial, the student is given the contents of all the tables for him or her to select from. Selection is made by entering the number associated with the data item. The program in turn displays the text associated with that data item and its pre-entered reply. THYROID also offers the student the choice of requesting a "consultant" to guide him or her through the differential diagnosis to the final correct diagnosis. Figure 3.4 shows an excerpt from a THYROID session.

{Having introduced the purpose of the system and the role of the student, the system begins by selecting one of the pre-entered cases}

YOU ARE CALLED TO SEE A 36 YEAR OLD WHITE WOMAN IN CONSULTATION IN THE HOSPITAL WARD. HER CHIEF COMPLAINT IS:

I HAVE BEEN FEELING QUITE LOUSY AND RUN DOWN LATELY.

ITEM No.: *305*
HANDSHAKE THE HANDS FEELS NORMALLY WARM AND DRY.

ITEM No.: *100*
DO HOT WEATHER OR HEATED ROOMS MAKE YOU UNCOMFORTABLE? NO, I LIKE THE SUMMER; COLD WEATHER, THOUGH, BOTHERS ME.

ITEM No.: *101*
ARE YOU EXCESSIVELY TIRED OR WEAKER THAN YOU USED TO BE? I FEEL TERRIBLY TIRED; I CAN'T DO ANYTHING ANYMORE.

Figure 3.4 - Excerpt from a THYROID session [Steele et al., 1978] (User input in Italics)

In terms of user interaction, this is the most primitive method ever seen in any of the medical CAI programs. While in terms of its overall usefulness, THYROID does not offer any real progress towards the development of medical CAL systems.

3.5.1.3 The MEDICS Program

The Swedish system, MEDICS [Hagglund *et al.*, 1981; Wigertz *et al.*, 1983], is a Branching Structured PMP simulation program for the training of medical students in clinical decision making. The program guides the student through a set of menus where each selected action results in displaying a string of text supplied by the simulation author. A simple scoring system based on pre-defined marks associated with each student action is used in this program. Figure 3.5 shows an excerpt from a MEDICS dialogue.

A 43 year old woman comes to the emergency room of the county hospital, where you have worked for 3 months as an intern in the surgery department, with a complaint of abdominal pain. The senior consulting physician is at home.

What do you do first? (Choose one alternative only)

1. Take a history
2. Do a physical examination
3. Order laboratory tests
4. Order X-rays
5. Order emergency room treatment
6. Admit the patient and begin definite treatment
7. Call the consulting physician

Alternative> *2*

Physical examination, management

Choose one alternative at a time from the list below. When you do not wish to choose any more alternatives, press "Return".

1. Skin	11. Abdomen
2. Peripheral Pulses	12. Rectal Examination
3. Blood Pressure	13. Proctoscopy
4. Heart Rate	14. Auscultation of Abdomen
5. Heart	15. External Genitalia
6. Lungs	16. Abdominal Reflexes
7. Temperature	17. Pupillary Reflexes
8. Lymph Nodes	18. Reflexes of the Extremities
9. Thyroid	19. Trendelenburg Test
10. Mouth and Throat	20. Laseque

Question>*3*: Blood Pressure 110/70

Question>*4*: Heart Rate 120

Question>*11*: Abdomen somewhat distended, no muscular rigidity. Severe tenderness in the lower quadrants - equal bilaterally. Rebound tenderness on palpation in the right quadrant.

Figure 3.5 - Excerpt from a MEDICS dialogue [Wigertz et al., 1983] (User input in italics)

In terms of medical education this system offers little over the standard PMP. However, from the computer science viewpoint, the main and probably the most important feature of MEDICS lies in the actual design of the system which was implemented in LISP. Data and control are defined separately. The database is frame-based. The control module consists of three basic types of data items: (1) Information Procedures which control the data retrieval process, (2) Action Procedures which are responsible for procedure invocation, and (3) Decision Procedures which are used for context changing. This modular design is rarely seen in any of the known medical CAI systems. This is partly due to the use of unsuitable programming languages (e.g. BASIC) or authoring languages (e.g. MICROTEXT).

3.5.1.4 The CASES program

A more sophisticated program than MEDICS has been developed in the Netherlands. This program, called CASES [Verbeek, 1987], guides the student through the basic three phases of hospital work: (1) Data Collection, (2) Disease Diagnosis, and (3) Treatment. During phase 1, the student is given a list of questions for him or her to select as if he or she is interrogating the patient. Appropriate replies are given to each chosen question. In addition, a VideoDisk is used to display pre-recorded medical illustrations when laboratory tests are requested by the student. In phase 2, the student enters his or her diagnosis for the case. If the diagnosis was wrong after a number of attempts, the program displays a list of plausible diagnoses for the student to select the correct one. Overall, the student should by the end of this phase have a clearcut view of the right diagnosis for this case. Finally, in phase 3, the student is asked to enter his treatment for this particular case. As in phase 2, the student has to give or be guided towards the same treatment specified by the simulation author. The positive feature in this system is the use of visual media as part of a CAI system.

3.5.1.5 Final Remarks

The main point that emerges from the description of the above four systems is the strict guidance on following the expert's line of decision making. This is not quite suitable for teaching medical students since, in real-life, doctors do not always get prompted on what to do next when faced with a patient problem. In addition, the strict guidance of these systems reduces their effectiveness in the actual teaching or tutoring process in this inexact science. That is, by following the expert's problem-solving strategies, the freedom of finding and evaluating any errors and misconceptions a student may have is restricted or on some occasions non-existent. The move towards total freedom in medical CAI systems is still to be made.

3.5.2 Intelligent Tutoring Systems

The potential of applying and developing ITSs in medicine is still to be explored in detail. ITS research has always been carried out by computer science and psychology researchers. The fact that most medical teaching/tutoring systems have been developed by medical personnel explains the lack of ITSs for this domain.

As indicated earlier, the teaching components are described as separate in an ITS. These components or modules are:

1. *Expert module* - Also known as the problem-solving module, its main task is the generation of problems and the evaluation of the correctness of the student's solutions. The expert module consists of a knowledge base of data and rules from a particular domain. The base is used to present topics for discussion, to perform inferencing on the student's input, or to match the student's answer with the expert's data. The expert component of an ITS is called an *articulate expert* as it can explain its problem-solving decisions in a similar manner to the explanations of a human expert. In contrast, expert components in other ITSs are known as *opaque* due to their inability to mimic the reasoning steps of human experts.

2. *Student Model* - The purpose of this model is to assess the student's knowledge in the particular subject area and hypothesize about his or her problem-solving strategies, explicit errors and probable misconceptions. The model is dynamically maintained.

3. *Teaching/Tutoring Module* - This module consists of a set of rules and procedures for the purpose of controlling the teaching activities. Its rules include knowledge on how to

teach, what to teach, and how often to interrupt the student. The knowledge in this module is usually domain-independent.

4. *Communication Module* - The purpose of this module is to control the input and output behaviour of the tutor. This module includes rules of discourse, coherency and rhetorical composition (e.g. suggest a topic only if mentioned by the student or logically related to the previous topic and then make reference to its earlier use).

Another problem with the few existing medical ITSs is concerned with the way these systems are built around expert systems [Clancey, 1982]. J.Self [1985] points out that greater emphasis is put on conducting the learner to the level of expertise of the expert system. Nevertheless, the main goal of teaching is not to transform each learner into an expert (an impossible task!). Rather, a good computer tutor is the one which clearly understands the students, adapting itself to the students' capabilities (the concepts they know, the strategies they use, and the misconceptions they have). This knowledge about the student forms the basis of the student model module in all ITSs.

The first pioneering hospital-based ITS is GUIDON [Clancey, 1987a]. GUIDON used a mixed-initiative dialogue and a case-method dialogue in its tutorials that were based on the information in MYCIN's knowledge base. MYCIN is an expert system for diagnosing and treating infectious diseases of the blood and meningitis. GUIDON's goal was to make the knowledge of the expert system accessible to students.

GUIDON's basic approach is to present the student with a case, supply additional information about the case when requested, and by keeping track of the requested information infer the problem solving method being used by the student. If this method appears inappropriate, GUIDON may interrupt to redirect the student's attention or perhaps teach him or her appropriate rule(s) from the knowledge base. GUIDON's teaching expertise includes capabilities to measure the student's competence and to use this as a basis for selecting what knowledge to present. The overlay student modelling technique is adopted by

GUIDON to represent the student's understanding. It also keeps track of what the student has learned in previous sessions and seeks opportunities (using the opportunistic tutoring technique) to broaden the student's knowledge of the medical data. Another feature of GUIDON is the use of "entrapment", which means forcing the student to make a choice leading to incorrect conclusions, thereby revealing some aspect of his or her (mis)understanding.

Clancey [1979] pointed out that the original GUIDON system failed due to the openness of medical diagnosis. That is, clinicians do not diagnose diseases using perfect recall on a huge set of medical facts and rules. They use factors common to several diseases to suggest potentially new dependencies, for example, yellow colouring is suggestive of liver disorders. Furthermore, the manner of the student-tutor interaction used in GUIDON is highly constrained in following the expert's correct solution path. This problem is still present in the new version of GUIDON.

During the 1980s, GUIDON was completely redesigned into a collection of instructional programs with added new features. GUIDON-2, the name given to this collection of programs, is based on the NEOMYCIN medical consultation system [Clancey *et al.*, 1984]. NEOMYCIN evolved as a reconfiguration of the MYCIN knowledge-base and EMYCIN, the domain-independent shell for building knowledge-based systems. NEOMYCIN has an enhanced explanation and tutoring capability. The new features in GUIDON-2 include:

- GUIDON-WATCH [Richer *et al.*,1988], a multiple windows graphics display front-end for use on a graphics workstation.
- GUIDON-TOURS [Barnhouse,1988] a program demonstrator for use by novice users prior to their work on GUIDON-WATCH.
- GUIDON-MANAGE [Clancey, 1987b] is a problem-solving environment in the style of ALGEBRALAND, where the student manipulates a set of operators whose detailed problem-solving effects are implemented by the system.

- GUIDON-DEBUG [Clancey, 1987b] is an experimental module that allows the student to modify the knowledge base by critiquing problem-solving sessions, for instance by suggesting an additional question.

3.6 EXPERT SYSTEMS IN MEDICAL EDUCATION AND TRAINING

Expert Systems can simply be defined as interactive programs which function as 'electronic consultants' in a particular domain.

Expert systems in medicine is an old-established research field with numerous implemented prototypes. Examples of expert systems in medicine include MYCIN [Shortliffe, 1976], INTERNIST/CADUCEUS [Miller *et al.*, 1984], ONCOCIN (a knowledge-based system for the management of cancer chemotherapy protocols developed at Stanford University [Shortliffe *et al.*, 1984], CENATUR [Aikins, 1983] and many more. The majority of medical Expert Systems have not actually been adopted for clinical use. In fact, most systems have been restricted to their original research centres. Despite published evidence [de Dombal *et al.*, 1986] of the benefits which may arise from using a computer system to support the human clinician in the making of certain clinical decisions, there is still some reluctance to accept the concepts of machine expertise and decision support. Part of this reluctance is due to unfamiliarity with the basic concepts of computers in general, and of expert systems in particular.

For the medical practitioner of today and the future, 'intelligent information processing systems' represent the best tool for handling the increasingly sophisticated large volumes of medical knowledge. The databases of most expert systems can be used in the teaching of differential diagnosis. The contents of such databases can be adapted to serve as a concentrated source of information that is more useful to clinicians than traditional reference sources. One such example is the large knowledge base of INTERNIST/CADUCEUS which contain over 3500 disease manifestations and covers over 500 diseases. Expert systems have been reconfigured to teach the principles of clinical management. An example of expert systems with instructional features is NEOMYCIN [Clancey *et al.*, 1984].

The need for such systems in clinical practice introduces a new element to medical education. Therefore, students must be aware of expert systems and their role in decision support.

For the students to get acquainted with expert systems two types of software are available for such introduction: (1) expert system shells, and (2) expert system teaching/training aids.

Shells are expert systems with their domain knowledge separated out - a shell, empty of knowledge. These shells provide an environment whereby the human expert can specify the facts and its associated rules to build the knowledge base and an interface to turn the knowledge base into the required expert system. Several currently available shells do not require any computer programming expertise.

Expert system teaching/training aids are programs similar to expert system shells but are less sophisticated and provide more instructions to the user than shells. The idea of expert systems teaching aids is implemented in a program, simply called *ESTA*, at the Department of General Practice, University of Liverpool [McWilliams, 1988]. This program models the behaviour of inductive expert systems.

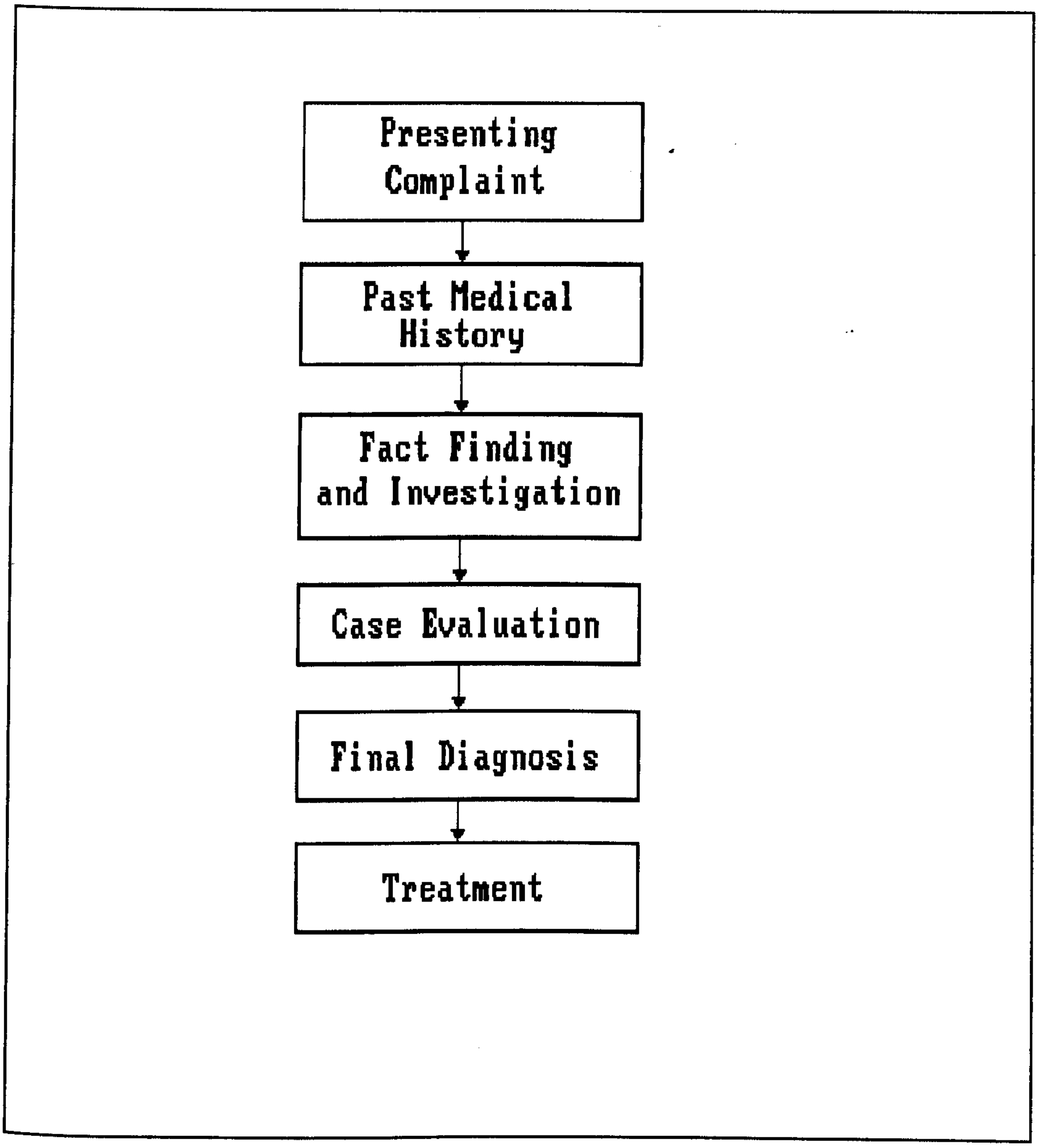


Figure 3.6 - Model of the Diagnostic Process as Commonly Taught at Present

3.7 CONSULTATION STRATEGIES FOR MEDICAL TUTORING SYSTEMS

Reviewing the documented and some of the commercial systems, two categories of systems emerge. These categories are based on the diagnostic model as (1) commonly taught at present, and (2) currently practiced. Figures 3.6 and 3.7 show the two models. The majority of systems apart from GUIDON are based on the common teaching method shown in the sequential model in Figure 3.6. GUIDON's strategy is based on the model shown in Figure 3.7.

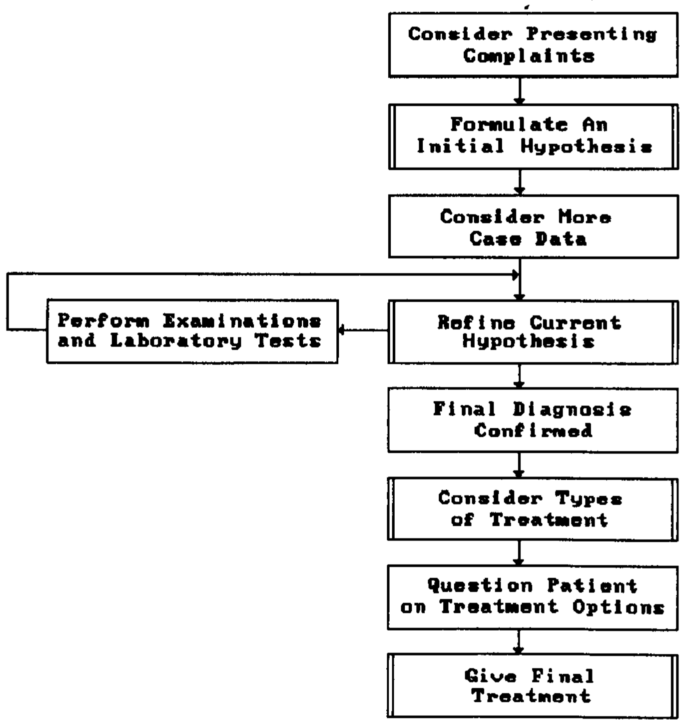


Figure 3.7 - Model of the Diagnostic Process as Currently Practised

The two models or structures have been adopted in all the medical systems whether CAI or ITSs and are ideal for hospital teaching where the emphasis, as mentioned earlier, is on disease diagnosis. This is different from Primary Care teaching where the emphasis is on patient management, communication, and observation skills. The different emphasis between the two types of care can also be seen when considering the patient role in both areas of care. In Primary Care, patients have problems. While in Secondary Care, patients are treated as cases with specific diseases or medical conditions.

Therefore, the model that most suits the Primary Care environment is shown in Figure 3.8. This model, which was devised in this joint project, clearly shows where the emphasis lies in this environment. The processes described in this model represents the working model of a consultation, or the decision making activity in a consultation. It could also hold true within the context of any medical consultation, not being exclusive to General Practice. Indeed this demonstrates one of the strengths of the *P.C.Tutor* system in so far as it simulates the data gathering protocols which are familiar to the largely hospital trained undergraduates while retaining the ability to simulate the more specialized working environment of the Generalist doctor. This model is the basis for our tutoring system.

The consultation model is displayed with full details to all the users of *P.C.Tutor* during their first session. The rationale for such action is for the students to shift their way of thinking from working in the Secondary Care environment to the Primary Care environment.

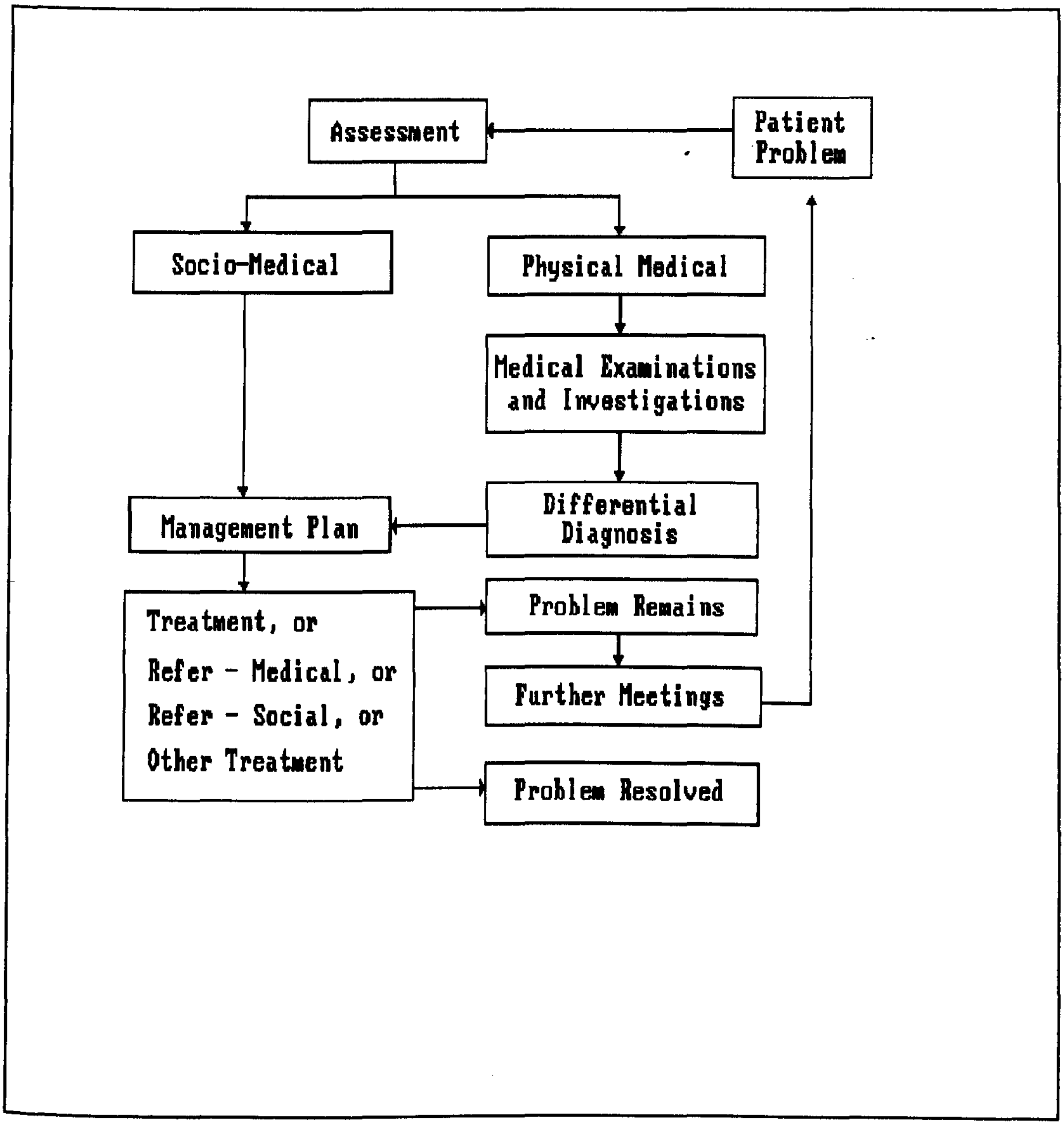


Figure 3.8 - Medical Consultation Model (A medical consultation starts with a presenting 'Patient Problem')

CHAPTER IV

DESIGN CONSIDERATIONS FOR A COMPUTER-BASED PRIMARY CARE LEARNING ENVIRONMENT

In this chapter, I will describe the software engineering method applied in the development of our medical tutoring system. I will also be discussing the design issues which are the basis of our tutoring system. In particular, I will be discussing the *Multi-Environment Simulation* method used in the *P.C.Tutor* project. Even though specifically set for a Primary Care environment, these issues can be applied to other domains with a similar problem-solving model. The issues or considerations for building the *P.C.Tutor* are:

- Database Design
- Data Entry System Design
- Simulating the Environment using the *Multi-Environment Simulation* Methodology
- The Provision for a Mixed-Initiative Environment
- Evaluation of Student Performance
- Provision for Access to the Medical Data
- System Modularity and Expandability

The overall objective of the above issues is the development of a framework for a learning environment for use in medical education systems, and in particular Primary Care systems.

4.1 SYSTEM DESIGN

Designing and implementing any type of computerized systems usually involves the activities: *Requirements Analysis*, *Specification*, *Design*, *Programming*, *Operation*, *Validation*, *Verification* and *Maintenance*. Such activities represent the classic life-cycle software engineering paradigm.

However, given limited resources (e.g. time, money, etc.), this set of activities become restricted and even difficult to achieve. Developing a medical tutoring system requires considerable time to specify the medical data and relationships in a way that can be manipulated by a computer (this process is commonly known as *Knowledge Elicitation*). Furthermore, designing a CAL system requires considerable time and effort in developing a human-machine interface. One of the features often associated with such interfaces is free-text data entry/processing (i.e. Natural Language Processing [NLP]). However, given the above constraints, and the difficulties associated with NLP, which is a research field in its own right, and also in the production of an interpreter for such a complex domain, NLP had to be omitted from the current *P.C.Tutor* project (Other issues regarding NLP are discussed in section 4.2.4). Because of these factors, the software engineering paradigm of *Rapid Software Prototyping* seemed the most obvious and suitable approach to use for its advantages over the other standard classic systems analysis and design paradigms or methodologies.

Rapid Prototyping is the "very quick" production of a working version of a piece of software. The emphasis, here, is on using whatever methods are available to produce either something that looks to the user like the final system or the actual final system. This software engineering paradigm has several advantages, where the important are summarized below:

1. During analysis, the system developers can show the user a suggested working system from a very early stage. This would enable the user to tell the developers how he or she feels about the system, and modifications can be made. The value lies in the fact that there is better communication between the user and developers thereby saving time.

2. During design, the greatest saving in time stems from avoiding the work in changing a system that is created in a conventional way which does not do what the user really wanted.

3. The user of the system participates in the design of the system.

The prototyping paradigm also has a greater advantage in terms of Human Machine Interaction (HMI) design. HMI is an important issue in the design of CAL systems where such an interaction plays a major role in communicating the system's knowledge to the student. It is difficult, if not impossible, to formally specify an HMI without the use of a prototype. Prototyping enables the HMI to be evaluated from the human perspective, using active participation rather than passive evaluation. Prototyping results in evaluation and the iterative application of the human engineering steps:

- *Activity Analysis* - The evaluation of all activities allocated to the human element in the context of required interaction with other elements.

- *Semantic Analysis and Design* - The definition of all actions and design of all the dialogue produced by the system.

- *Syntactic and Lexical Design* - The identification, representation and design of all actions and commands in the HMI interface.

- *User Environment Design* - The combination of all hardware and software elements forming the user environment.

In the *P.C.Tutor* project, the role of the user was played by the General Practitioner working on the CTISS project.

Prototyping has been categorized into three categories: *Throw-away*, *Evolutionary*, and *Incremental* [Hekmatpour *et al.*, 1986]. Of the three categories, the *Evolutionary prototyping* approach was used in the development of the medical data entry and tutoring systems. In this approach, a system grows and evolves gradually. New additions and modifications can easily be made in this design approach. Further, the approach encourages the development of highly modular designs, an important issue in the development of expandable systems.

The Rapid prototyping paradigm is sometimes attributed to producing "dirty" designs. Such designs are usually the outcome of the *throw-away* prototyping method. Here, the purpose of the prototype is to show how parts of or the complete system will look. Once the prototype has been demonstrated it is thrown away and a new system based on it is developed and tested using either the classical methodologies or the other prototyping techniques.

Using the prototyping paradigm requires the use of suitable application or software factories (e.g. very high level programming languages (VHLL), 4GLs, Authoring systems, etc.). Examples of VHLLs include LISP, Prolog, Smalltalk, etc.

In terms of CAL, the nature of the CAL development language can affect the character of the CAL material, and it also has implications for the way in which the CAL system is designed. The use of VHLLs during the initial phase of the design process enables designers to readily prototype ideas for development which are linked to the powerful ideas and cognitive model in the particular language. This has implications for the structure of the working model that is adopted for development. In particular, the initial phases of the model take on quite a different character, with the role of the programming language becoming far more influential. Figure 4.1 attempts to represent these changes.

As can be seen from Figure 4.1 the initial phases of the design of CAL software could be significantly affected by prototyping within a powerful system. The development system acts as an environment for the teacher/designer to try out educational ideas. This could have the effect of changing the nature of the development that the software is designed to sponsor, as the cognitive model inherent in the prototyping language will affect the style and content of the software that is developed.

Instances of languages being used in this fashion are not very common. This may be because of the absence of efficient implementations of VHLLs on 8-bit computers which are commonly available in schools and universities. However, this situation is changing in two ways:

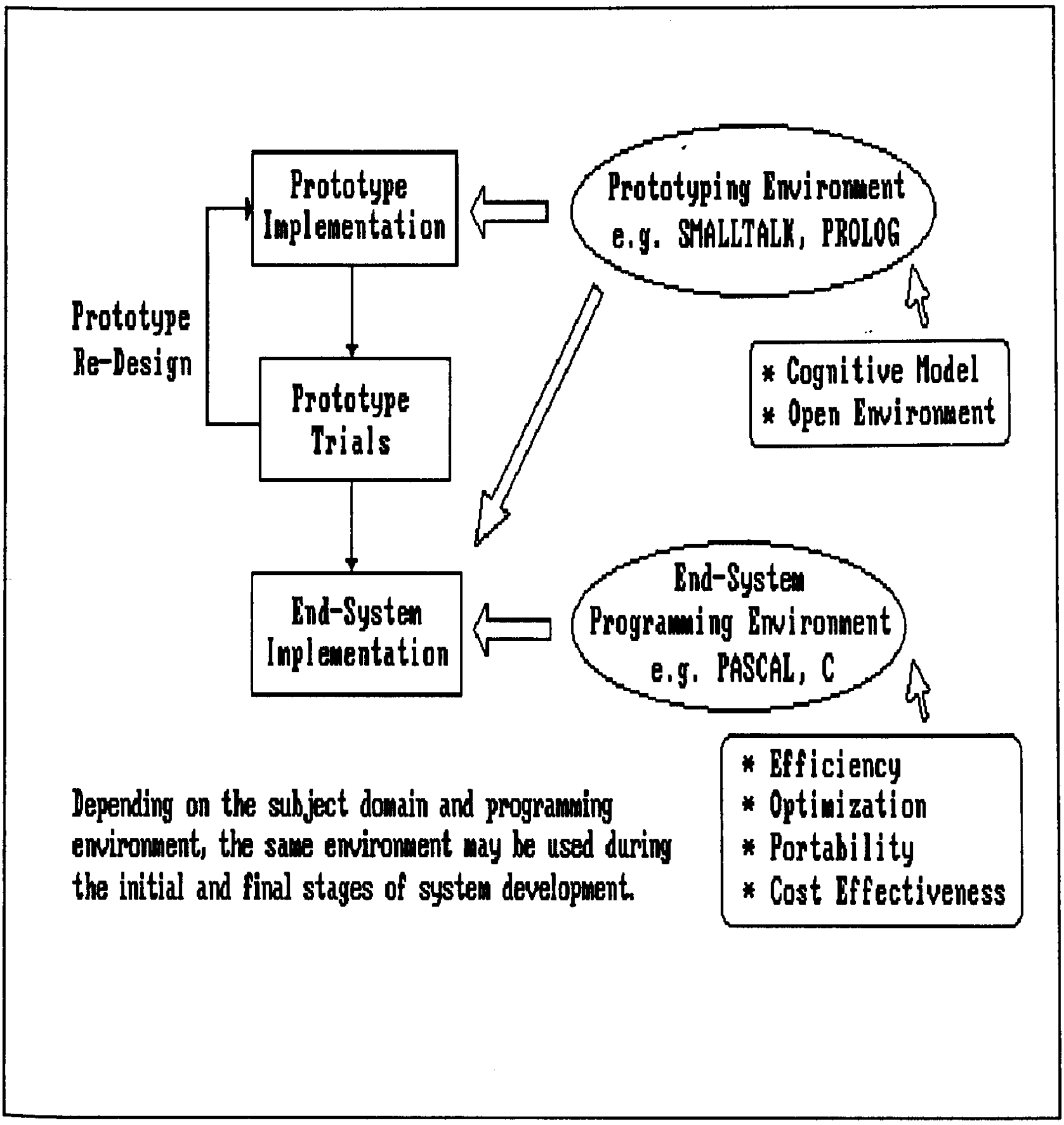


Figure 4.1 - Rapid Prototyping of Educational Software

1. Powerful 16 and 32-bit computers with fast processors and enhanced memory are becoming available at reasonable prices.

2. More powerful and sophisticated programming environments are also becoming available on such machines.

P.C.Tutor was developed in the Turbo Prolog(TM) environment. The use of Prolog in prototyping was very limited until few years ago when a number of sophisticated and affordable programming environments were developed for use on personal computers. An additional piece of software, called DEMO (TM) was also used prior to the development of *P.C.Tutor* for simulating the interface and screen layouts designs. The DEMO system is a software presentation and prototyping package. A realistic view of the final system was built in the form of slides and run as the complete system. Once the designs of the HMI in the prototype were accepted, the prototype was "thrown away" and full development of the tutor was initiated on that basis.

4.2 DESIGN ISSUES

4.2.1 Database Design

The application of relational database design in all types of CAL is confined to very few systems (e.g. STUDIO [Grahne, 1981]). The data structures that are commonly used in CAL systems are *frames*, *scripts*, *production rules*, and *semantic networks*. These structures are usually stored as knowledge-bases consisting of domain facts, rules specifying how these facts are related to each other and rules for manipulating the facts and their relations. Often, in such systems, an interpreter is required for the translation of this knowledge into tutoring operations. This is totally appropriate for one-task systems (i.e. systems whose purpose is to *teach* a particular subject). Yet, for a system to offer more than tutoring (i.e. one that can be expanded to include other tasks such as expert assistance, reference book, etc.), some or all the knowledge other than the raw data becomes obsolete.

One of the problems that can be attributed to the lack of CAL systems built on relational database designs is the unsuitability of most data base management systems (DBMSs) for educational purposes. Yet, the advantages of a DBMS in handling large amounts of data and complex relationships sets up the basis for a comprehensive learning environment offering the student the above features and many more. The only problem with a DBMS-based system is the need for a development environment that is capable of implementing such design.

The DBMS approach is used in *P.C.Tutor* where the bulk of the raw medical data is stored as Prolog facts in a separate file. Access to this data is made using specially built indexes. All relations and rules are stored as part of the source code in the form of production rules. In Turbo Prolog, the internal database is stored as ASCII code. Therefore, the rationale for explicitly defining the rules within the source code is to prevent any person other than the system supervisors from modifying their contents due to the fact that some of these production rules are used by the student model. The DBMS design was successfully implemented by making full use of the Turbo Prolog External Database system [Turbo Prolog Users Guide, Borland Inc., 1988].

This simplified, structured design method gives the student the ability to learn from browsing, scanning, and exploring the database without the need to view any rules as is the case in other systems.

In addition, the modular structure has the advantage of clearly organizing the information and in making the editorial process simpler. Further, a highly modularized system could be developed on the basis of the *P.C.Tutor's* medical database as in the example shown in Figure 4.2.

The following sections discusses the structure and content of the medical database.

4.2.1.1 Database Structure

A General Practitioner requires three types of information for solving a patient problem:

1. *Background knowledge* about the patient (e.g. past medical history, personal history, social history, family history, etc.), and the community (e.g. social factors, economic factors, etc.).

2. *Clinical information* obtained from the patient in the form of presenting complaints and observed by the doctor through any examinations.

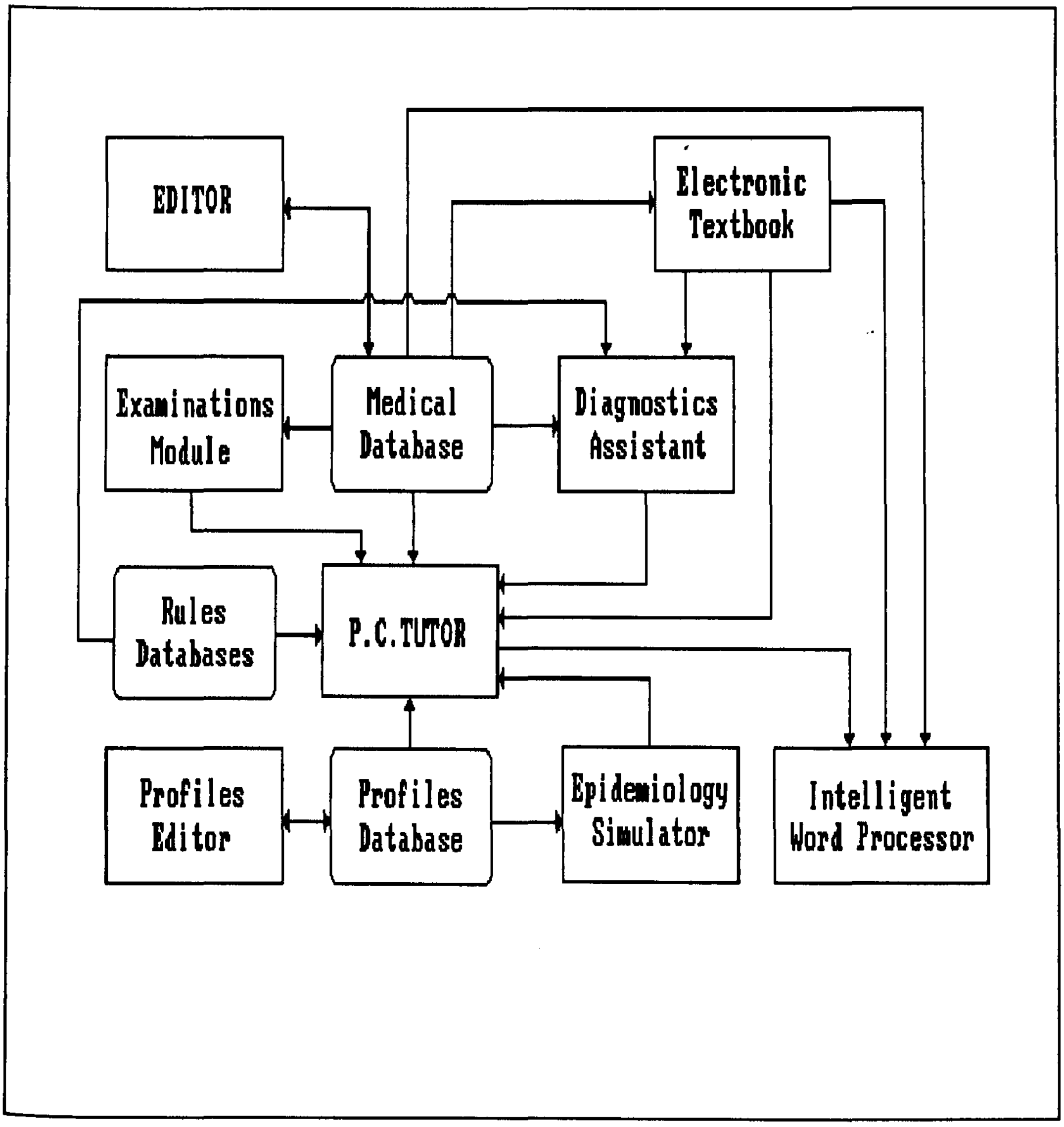


Figure 4.2 - A Block Architecture for a Learning Environment for use by Students of General Practice and General Practitioners for Vocational Training

3. *Medical knowledge on disease entities* (i.e. the full facts on all diseases and medical conditions).

This information is not only relevant to a General Practitioner but also to a hospital consultant. A tutoring system that aims to simulate a Primary Care environment must cater for this information in order to attain a realistic simulation. The majority of medical teaching systems have as part an authoring system for entering such information (usually in the form of case histories). In our system, we opted for a different approach, that of a simple Prolog database.

The medical database is simply a collection of Prolog facts stored on disk and accessed in sequential and random (using special indexes or more specifically B+Trees) orders. Each disease is represented by one of these facts. The current version of the tutoring system consists of a database of some thirty disease and other medical condition entities. Background knowledge concerning simulated patients is to be implemented in the next version.

The medical data stored in this database consists of specific diseases and medical conditions identified by the General Practitioner to be the most common ones that are encountered by the doctors in this country. The rationale for the simple database approach is based on the following factors:

a. By using disease-specific data, time is saved by the person entering the data. Authoring systems used with other teaching systems require considerable time on the part of the doctor to enter a single case;

b. System expansion is made simple. Adding more data types is a matter of increasing the number of parameters in the Prolog facts.

4.2.1.2 Database Content

The medical database consists of the following items of data:

1. The disease name (e.g. Rubella), its synonyms (e.g. German Measles is another name for Rubella), and the category the particular disease is part of (e.g. Infective and Parasitic Diseases, Diseases of the Respiratory System, etc.). The diseases and medical conditions are grouped into 19 categories or groups as divided by Hodgkin [1985]. These groups

were found to cover all the diseases and medical conditions encountered by a General Practitioner. These categories are also used in the structuring of the database.

2. The symptoms experienced by the patient when attending the surgery. The symptoms are given to the student in the form of sentences that simulate a real-life patient-to-doctor dialogue, i.e. they do not include any medical jargon (e.g. *Moderate sore throat for the last 3 days*). The symptoms in *P.C.Tutor* are classified as *Presenting* (i.e. Symptoms which represent the main features of the particular disease or problem) and *Non-Presenting* (i.e. Symptoms which are less common that are experienced with the particular disease or problem. These are not always perceived by all patients). Appendix A shows the method used in the organization of the symptoms hierarchy.

3. The clinical signs (non-laboratory) that can be identified by the General Practitioner in order to help in the assertion of the existence or non-existence of a particular disease or condition. The signs are usually identified through observations or simple examinations at the surgery. In contrast to the symptoms, the signs use medical terms and expressions (e.g. *Erythematous maculo-papular rash of the chest*). For details on the structuring of the clinical signs, see appendix A.

4. The laboratory tests necessary to positively affirm the existence or non-existence of a particular disease or condition to the General Practitioner. The laboratory tests are in two parts: (1) Tests which are performed by the General Practitioner at the surgery (mainly Urinalysis tests), and (2) Tests that are performed at outside laboratories (e.g. X-Rays, Blood tests, etc.).

5. Any interventions or treatment procedures that are necessary for such a disease. These interventions and treatment procedures represent the management plan for the disease or medical condition. For each disease, there are four lists of actions associated with it. The four lists are classified as:

a. *Important actions list* - this is a list of actions that are vitally important and should always be followed by the student (e.g. Giving simple analgesics to patients with Glandular Fever for symptomatic relief).

b. *Optional actions list* - such actions can be optionally taken by the student as they do not bear any negative effects on the patient (e.g. giving simple analgesics as a temporary treatment). For example, giving Penicillin to a patient with acute sore throat suspected to have either Glandular Fever or Tonsillitis.

c. *Negative or harmful actions list* - the actions in this list should always be avoided when managing a case where the particular disease is a possibility (e.g. a GP should never give Penicillin Esters to a patient with acute sore throat. This could be lethal if the patient has Glandular Fever).

d. *Unnecessary actions list* - these actions, though not harmful to the patient, are considered by most General Practitioners to be unnecessary when managing a particular problem. These actions are usually considered unnecessary because they are costly and time consuming to both the patient and the Practitioner. For example, it is unnecessary to give Penicillin to a patient with simple sore throat since it has no benefit to the patient. However, this may be argued to be an acceptable treatment where diagnosis is in doubt and the symptoms are severe.

6. The distribution of a disease in the community (i.e. disease prevalence). For example, the national prevalence for Influenza is 10 in 1000 patients in normal conditions (i.e. not during an epidemic).

7. The age incidence for the particular disease. The age is represented as a numeric range.

8. The sex incidence that is relevant to the disease. This is represented in the database as a ratio (e.g. the sex incidence of Inguinal Hernia in males to females is 10 to 1).

9. The stages through which the disease usually goes. The data on the life-span of a disease is partially implemented in the present version of *P.C.Tutor*. The complete life-span details of a particular disease will be implemented at a later stage in the development of the medical system. The stages of a disease will include data such as new symptoms, clinical signs, the type of treatment needed for the particular stage, type of drugs, any negative actions including explanations (if any), the time taken for this stage to finish, and many more.

Appendix B shows the current content of the medical database.

4.2.2 Data Entry System Design

A simple data entry system has been developed to create, modify and extend the medical data stored in the medical database.

The *Editor*, the name given to this system, has been specifically developed to enable a General Practitioner or any other medical person to clearly and in an organized manner specify the medical data required for building the medical data base. It is a simple database editing system offering all the facilities of an information storage system.

The *Editor* is a menu-driven user-friendly system. It is designed for use by any person with no computer programming experience.

In contrast to authoring systems, the *Editor's* purpose is to organize the raw medical data into pre-defined database structures. The data is entered without any rules governing its manipulation by the tutor. Such rules are either inferred by the tutor from the data or specifically written in the tutor's source code.

4.2.3 Simulating the Environment using the Multi-Environment Simulation Methodology

The large number of medical CAL systems (including the systems discussed in the previous chapter) that are documented in the medical journals claim to be based on the simulation-type teaching strategy. Yet, these systems when closely observed reveal a similar underlying simulation pattern. This pattern is based on the way the physician works in his or her environment. Most of these simulations do not take into account any of the outside influences such as changing life-styles, employment problems, and many more.

In Chapter 3, we considered the problems with known teaching systems aimed for use in Primary Care undergraduate teaching. The approach that I have taken, which was confirmed by the General Practitioner working on the CTISS project, was to stress the tutor's emphasis on patient management rather than disease diagnosis. In this case, the patient is seen as a whole rather than a collection of interlinked sub-systems as in most CAL systems. This unique approach has the advantage of closely simulating the Primary Care environment as well as the problem solving method used by all General Practitioners. *P.C.Tutor* simulates individual patients who present to the student in the same way that patients present in any General Practice. The system produces simulated patients in case mix which mirrors the actual incidence of disease entities within the population being simulated so that the "surgery" of patients seen by a student in each tutorial has a realistic feel.

The simulation of problem-solving in General Practice was achieved by permitting the student to consider any number of diseases in his or her hypothesis and suggest a management plan on the basis of this hypothesis.

In *P.C.Tutor*, the student's goal in any tutorial is to correctly identify the tutor's selected disease or any subset of diseases that could account for the presented information (e.g. symptoms, signs, etc.), for the purpose of managing the particular case. In General Practice, cases are diagnosed either:

a. on the basis of personal experience, patient history, and/or clearly identified symptoms and/or signs; or

b. for temporary management, if more than one disease is being considered by the General Practitioner (i.e. his or her hypotheses about the case in hand). Hodgkin [1985] points out that General Practitioners usually face a high proportion of trivial problems and, therefore the use of hospital diagnostic techniques, which involves ordering complex investigations, are unjustified. As a result, in most instances, General Practitioners allow time to solve cases. Based on previous knowledge, the doctor identifies the likely trivial causes and how long they should persist. The patients are told what to expect and informed that they must return if the symptoms have not cleared up in the predicted time. The length of time the practitioner waits will be decided by the natural history of the possible serious disease suspected. The doctor may wait a few hours or several weeks.

The tutor aims to teach in a complementary way the medical facts on all diseases encountered in such an environment and the problem-solving strategy associated with each disease. It does not attempt to teach any communication and observation skills. Such skills are unlikely to be acquired from a computer system.

In general, for a tutoring system simulating the Primary Care environment to be effective in teaching students procedural and heuristic knowledge in addition to the laws and principles underlying the sets of relationships modelled by the system, it should be based on the general consultation model shown in Figure 3.8.

A second significant issue in the design of a simulated Primary Care environment is concerned with the type of community the particular system is modelled on. A system of this nature should also have the ability to simulate different Primary Care environments (e.g. Urban, Rural, Mining, etc.) where each one has its own health and social problems (i.e. the provision for a *Multi-Environment Simulation*). A multi-environment simulation system offers the student greater knowledge expansion by enabling him or her to manage cases which are rarely seen during the clinical training (e.g. simulating an environment in Africa where diseases such Malaria, Sickle Cell Anaemia, etc., are common). The multi-environment simulation method in *P.C.Tutor* was achieved by modifying the data values in the appropriate community profile databases. These databases, implemented as internal Prolog databases, include the *population*, *consultation*, and *disease* (or *problems*) profiles. The *population* profile represents the overall working General Practice environment (e.g. Urban, Rural, etc). The *consultation* profile is a representation of the patients visiting the simulated surgery. Finally, the *disease* profile represents the disease incidence. Apart from closely simulating the environment, by varying the parameters in any one or more of these profiles, it is possible to simulate disease epidemics. Epidemiology, the study of disease epidemics and their effects on the population, is one of the important subjects in the medical domain. Studying disease epidemics is always difficult since a student can only, most of the time, read about it in the textbooks. Therefore, providing the ability to simulate disease epidemics is one of the important design features of the *P.C.Tutor* project.

4.2.4 Providing a Mixed-Initiative Learning Environment

Educational software can be classified into three categories: Computer Controlled, Mixed Initiative (referred to as *Mixed Control* by some authors), and Learner Controlled. Computer Controlled software (e.g. Drill and Practice, CAI) have been criticized by researchers in the AI and medical fields as highly restrictive and unrealistic in student teaching [de Dombal, 1971]. In the second type, Mixed Initiative, which include simulation and educational games software, the initiative is shared between the student and the program. Such programs are favoured by most educationalists since the student is encouraged as far as possible to think for him- or herself and to develop his or her own initiative. The last type, Learner Controlled, include all general purpose programming languages, fourth generation languages, data base systems, word processors designed for students, music editors, and so on. Although such systems are of great value to student learning, they in fact place quite a burden on teachers to provide the support needed by the students, especially novice students [Riel *et al.*, 1988].

In a mixed-initiative system, the student is given the control over the system most of the time. GUIDON is the best example of a mixed-initiative medical tutoring system. Designing a system using this learning strategy prompts a decision on the implementation method. Two methods are available: (1) the use of Natural Language Processing (NLP) and (2) the use of keywords.

NLP is a major research area in the AI field. Designing an NLP system requires the consideration of the following issues:

1. Time available to the project.

2. Human expertise in NLP.

3. Hardware and software resources available to the project.

4. Software support for NLP.

5. Time and processing speed of an NLP system as opposed to a keyword-based entry system.

6. End-user view of NLP.

The majority of NLP-based systems are still as prototypes or as research systems.

A keyword-based system, on the other hand, has its advantages and disadvantages. The advantages include faster implementation and processing time (less time is taken by the student to work through a case), and greater software support. The main disadvantage is the restrictions imposed on the student by the limited number of options or keywords. Taking into account the short time available for medical students for formal tutorials and the prototyping approach applied to the development of *P.C.Tutor*, a keyword-based system seemed the appropriate design choice for our system.

Another critical issue for consideration when designing mixed-initiative tutors is the tutor interaction with the student. The amount of feedback to the student during the course of the interaction varies with the different tutoring systems. Harless [Harless *et al.*, 1971] believes that to sustain the reality of the simulation and to provide each student with the opportunity to engage in meaningful, independent decision making, there should be no pedagogical interruptions while the problem evolves under his or her control. From the initial patient description until he or she decides on a possible diagnosis and prescribes an appropriate treatment, a student should proceed entirely in his or her own style of information gathering and problem solving.

A different view was taken by de Dombal [de Dombal *et al.*, 1969] in his program which has a much more active control over the student-tutor interaction in that the student is forced to collect all the relevant information in each subset of the problem: typical subsets are characteristics of the presenting complaints, associated symptoms, past history, etc.

B. Woolf [1988] points out that in designing tutor responses, we need to consider how much information to give the student in the interactive presentation, and what motivational comments to make; for example whether or not to:

- acknowledge the student's correct responses;
- give the student the correct response or answer;
- provide a justification on all the correct and incorrect responses of the student;
- give hints or leading questions;
- challenge the student's answers with counter-suggestions; or

- provide additional information which extends the content of the current question-answer interaction.

Motivational factors may include encouragement, congratulations, challenges, and other statements with effective content.

The method of student-tutor interaction can be explained as follows. To build a medical tutor that can be used to teach students of General Practice how to logically use their decision making processes in managing a patient problem, one needs to define the areas where a student-tutor interaction is required. These areas are:

1. The collection of the medical history of a patient. This involves examining the patient record and the symptoms that he or she has. In the present version of *P.C.Tutor*, patients are affected by one disease only.*

2. The clinical examinations, signs and some of the laboratory tests that can be performed by General Practitioners.

3. The formulation of a differential diagnosis (i.e. the list of diseases considered by the tutor or student to have similar presenting case data).

4. The management plan relevant to the particular patient and problem under investigation.

Referring back to Chapter 3, Figure 3.8 also represents the main areas of student-tutor interactions. In our tutoring system a student considering a Socio-Medical problem is offered the same set of procedures that are available to him or her when dealing with Physical-Medical condition. Yet, in real-life, although Figure 3.8 shows the sequence of events/actions taken by all General Practitioners, some deviations or shortcuts are sometimes made depending upon personal experience and clearcut symptoms and/or clinical signs. The student-tutor interaction provided in the *P.C.Tutor* system follows to a certain extent the points

* This situation will be changed in the next version where past medical history information will be taken into account.

expressed by Woolf [1988]. The present system is somewhat restricted by the limited information built into the database. For example, details regarding the relevancy of some of the data items (clinical examinations or laboratory tests results) are not included, thus restricting the question generation and explanation capability of the tutor. The inclusion of such details and others are planned for the next version of the system.

To summarize this design issue, a "good" tutoring system is one where the tutor plays the role of a knowledgeable friend who only suggests and advises, leaving the control of the learning situation essentially in the hands of the student.

4.2.5 Evaluation of Student Performance

Modelling student understanding of a subject domain has always been one of the important aspects of ITSs. As Clancey [1987a] pointed out that:

> "We generally think of a student model as an evolving description of what the student knows about the subject domain.From this perspective, we start with a corpus of facts and relations, encoded as rules or a semantic net, and *paint in red* what we think the student knows."

Great effort has been made in the study of the student model. Student models used up to now in tutoring systems can be classified roughly into four categories: *global measures of performance*, *overlays*, *bug-libraries*, and *runnable models*.

In the overlay model, the student's knowledge is represented as a subset of the expert's knowledge. It is a practical technique since its modelling mechanism is very simple. But it has a serious restriction on the modelling capability, since it cannot represent the student's incorrect knowledge.

The bug-libraries model method synthesizes the student model from a prepared set of knowledge including both correct and incorrect conceptions. If the student's knowledge has complex structure and includes many misconceptions, the system needs heuristic knowledge for finding out which correct conception should be replaced with a wrong one, because a general problem solver is not defined for the bug model approach. Performance of this method depends on how well a prepared set of knowledge is adjusted to the teaching material.

The bug model is a kind of knowledge-based system and gives a practical approach to student modelling at the current state of the art. A kind of knowledge-based system is efficient when the knowledge base is rich. However, if the knowledge base is incomplete, it sometimes fails to solve a given problem because it does not have a general problem solver.

The fourth category, the runnable model, is most suited to complex domains. In this model, the system not only detects which correct concepts and bugs the student uses but also which goals he or she wants to achieve, how he or she uses plans, and how he or she combines concepts into more or less correct solutions of a problem. Examples of systems based on runnable models include NEOMYCIN [Clancey *et al.*, 1984], and the Pascal-Tutor PROUST [Johnson *et al.*, 1987].

In chapter 1, I outlined the student model used in *P.C.Tutor*. The dynamic process, pointed out by Clancey and described below, is based on various sources of evidence and ways to reason about the evidence. Such evidence is discussed in the sections that follow with details on its application in *P.C.Tutor*.

4.2.5.1 Sources of Evidence

The evidence for updating the student model are classified into three groups: **Background** evidence for initializing the model, **Implicit** evidence, and **Explicit** evidence [Carr and Goldstein, 1977; Clancey, 1987a].

Background Evidence

At the start of the first tutorial, all students are considered to have the same level of experience. This estimation of the student level of understanding (or experience) is updated during the tutorial session.

Other background evidence is also incorporated from the second tutorial session and onwards. This evidence includes the names of diseases managed and used by the student during the previous sessions.

Implicit Evidence

Indirect clues can be used to reason about the knowledge guiding the student's behaviour. In *P.C.Tutor* this implicit evidence is derived from the student's hypotheses and his or her choice of options. In general, the data the student requests and the forms of assistance he or she requires is used to deduce his or her focus of attention and whether he or she is struggling to solve the problem. As a rudimentary form of this aspect of modelling, when the student requests help for solving the problem (i.e. requesting HELP during hypothesis entry or requesting case data), the program lowers its belief that the student knows the particular part of the subject domain that is selected for presentation.

Explicit Evidence

A tutor can also find out what the student knows or is thinking about by asking him or her directly. In *P.C.Tutor*, questions are constructed from the information given to the student. Two types of questions are used by *P.C.Tutor*; multiple choice and simple true-false questions. Student's responses to the questions are assessed and the model is updated accordingly.

4.2.5.2 Components of the Model

The student model consists of two components: The **Background** and the **Current Model** components. Figure 4.3 shows how the components of the student model are related to each other and to the sources of evidence for modifying the model. The Background component is the only component that is carried from one session to another for each student. The Current Model component is started up using the information from the Background component and continuously updated during the session.

The implementation of the two model components are explained in detail in the next chapter.

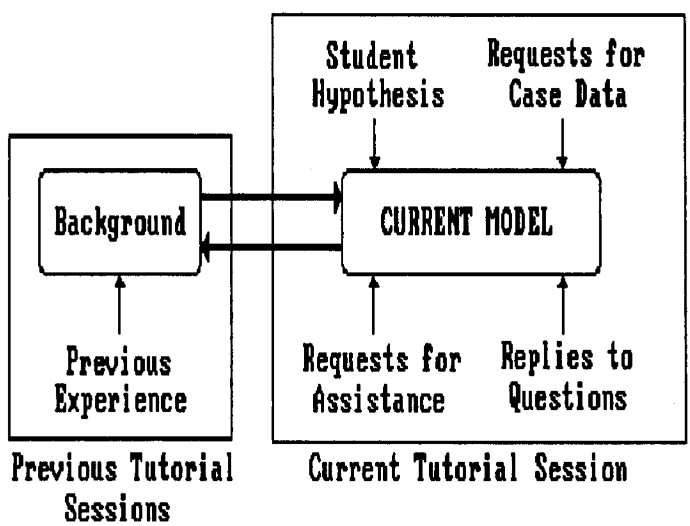


Figure 4.3 - Components of the Student Model and Sources of Evidence

4.2.6 Provision for Access to the Medical Data

Providing access to the knowledge used by the teaching system is an issue which is considered and implemented in very few ICAI systems, to be more specific those which use an expert system for tasks generation. In current medical CAI systems this facility is not made available to the students mostly due to the fact that this knowledge is explicitly defined in the source code of these systems. This provision for access can be considered as a knowledge retrieval system or an electronic reference- or textbook. This feature is an important design issue for future medical CAL and expert systems.

In the present version of *P.C.Tutor*, the student is offered the option of examining the contents of the medical database at the end of each completed case and during his or her entry of incorrect disease(s).

Additional processes have been added to the *P.C.Tutor* system offering the student:

1. The ability to cross reference the diseases and other conditions in the database by the presenting and/or all symptoms. The symptoms are selected by the student using a hierarchical set of menus.

2. The ability to cross reference the diseases, and other conditions in the database by management actions. As in (1), any set of actions can be selected from a number of multi-level menus.

Rapid searching and cross referencing routines are very suitable in the field of medicine where such processes are time consuming when performed manually. In fact, this issue is the main reason for the development of electronic textbooks based on some of the well known medical reference- or textbooks. One example is the Oxford Textbook of Medicine by the Oxford University Press.

Details of future expansion on the basis of this issue is considered in Chapter 7.

4.2.7 System Modularity and Expandability

The literature on software engineering always stresses the importance of modularity in systems design. Modularity allows the replacement and reassembly of individual modules without the reassembly of the whole program. It also allows one module to be written with little or no knowledge of the code of other modules. A module may be considered to be a subroutine or a collection of subroutines that are interdependent, so that the sequence of instructions that is necessary to call a given routine and the routine itself are part of the same module.

A modular system has the advantage of easier maintenance and greater scope for expansion. To examine the appropriateness of this issue one needs to look, in brief, at the design of CAI systems and ITSs.

The vast number of CAI systems built using authoring systems or other systems generators have one aim only and that is to provide computerized instruction in problem-solving. None of these systems has the ability to act as a learning environment whereby the student is in total control of his or her learning. Furthermore, customization of tutorials, where the student is given at least some control of his or her problem-solving method, is always difficult due to the restrictions imposed by the authoring system. CAI systems built using an authoring system always have the "black box" design as opposed to the "glass box" design where the teacher and the designer have the ability to examine and manipulate the contents of the system. Therefore, in the case of a "black box" design, expansion is difficult, if not impossible.

Early ITSs which were built from scratch without the use of an authoring system were both highly complex and unmanageable (WEST [Burton *et al.*, 1982], SOPHIE [Brown *et al.*, 1982], GUIDON [Clancey, 1987a; Clancey *et al.*, 1988]). Although ITS authors profess to have organized their code into three distinct modules (namely the expert, student and tutoring modules), the authors freely admit that this is an abstract notion of how the tutors were actually built [Woolf, 1984]. Most of the code for the different modules are scattered throughout the systems. In general, early tutors were not highly modular. Due to this, some of these systems were completely redesigned as in the case of GUIDON-2 (see section 3.5.2).

As indicated earlier in this chapter, evolutionary prototyping requires highly modular design. In *P.C.Tutor* modularization was achieved by closely following the processes shown in Figure 1.2 and the way the medical data was represented in the database (e.g. symptoms, clinical examinations, laboratory tests, management actions, etc.).

P.C.Tutor was built around the general consultation model shown in Figure 3.8. In adopting this model, one can, for example, redirect the emphasis from patient management to disease diagnosis for use in hospital teaching. Further, the use of menus in *P.C.Tutor* enables any computer designer/programmer to add new options (or modules) for the student to access during a tutorial.

CHAPTER V

SYSTEM IMPLEMENTATION

5.1 THE MEDICAL DATA ENTRY SYSTEM

The data entry system has been rapidly prototyped in Turbo Prolog V2.0, for its speed, screen handling, and database management. Turbo Prolog has an efficient compiler which generates standard object files, hence enabling the addition of similar files generated by other languages. The medical database is represented as Chains, structures which are similar to linked lists but with the added flexibility of relational database systems [Turbo Prolog V2.0 Users Guide, Borland Inc., 1988]. The main feature of the Chains structure is the use of indexes (or more specifically B+ trees) for the random access of each element (or term) in the particular chain. The Editor system consists of a collection of programs:

1. The medical data entry program. This program, the main one in the system, is used for the addition of new and editing of existing diseases and medical conditions (this program is appropriately called the *Editor*).

2. Community profile databases creation and maintenance program. As explained in chapter 4, these databases are used for simulating the different General Practice environments in any part of the world and even out-patient clinics in hospitals. Using the Multi-Environment Simulation method, the teacher can broaden the scope of the tutor and help prepare the student for almost any medical environment.

3. The indexes creation program. Special indexes have been built for all the data stored in the medical database. The purpose of such indexes is to rapidly search and access the medical data. Some of these indexes are currently used by the tutoring system, while the rest are kept for future use.

4. Two housekeeping utilities for producing hardcopies of the medical data and making back-ups of all the data files on floppy disk.

The data entry program, the *Editor* in point (1) is the most important of all the above programs and therefore it will be explained in more detail.

5.1.1 The Editor

The main function of the *Editor* is the creation and maintenance of the medical database. The program consists of a number of procedures, each of which has its own data entry screen(s). All procedures make full use of specifically defined function keys and menus for quick data entry. An estimated time for entering one disease is approximately 20 minutes as opposed, for example, to the 10 hours required for writing a case history in the Glasgow system [Murray, 1977]. These procedures are outlined in the sections below.

5.1.1.1 The Disease Life-span Progress Entry Procedure

Teaching a student the management of patients in a General Practice environment requires the simulation of all the stages the particular disease goes through. The majority of diseases have been found to fit into four types of disease progression graphs (Figure 5.1). The four graphs represents:

a. Self limiting diseases (e.g. Influenza, Measles, etc.).

b. Diseases that remain at a constant rate (e.g. Inguinal Hernia, Acne Vulgaris, etc.).

c. Diseases that continue to worsen with time (e.g. Senile Dementia, Carcinoma of the Bronchous, etc.).

d. Diseases that worsen with periods of remission (e.g. Multiple or Disseminated Sclerosis, etc.).

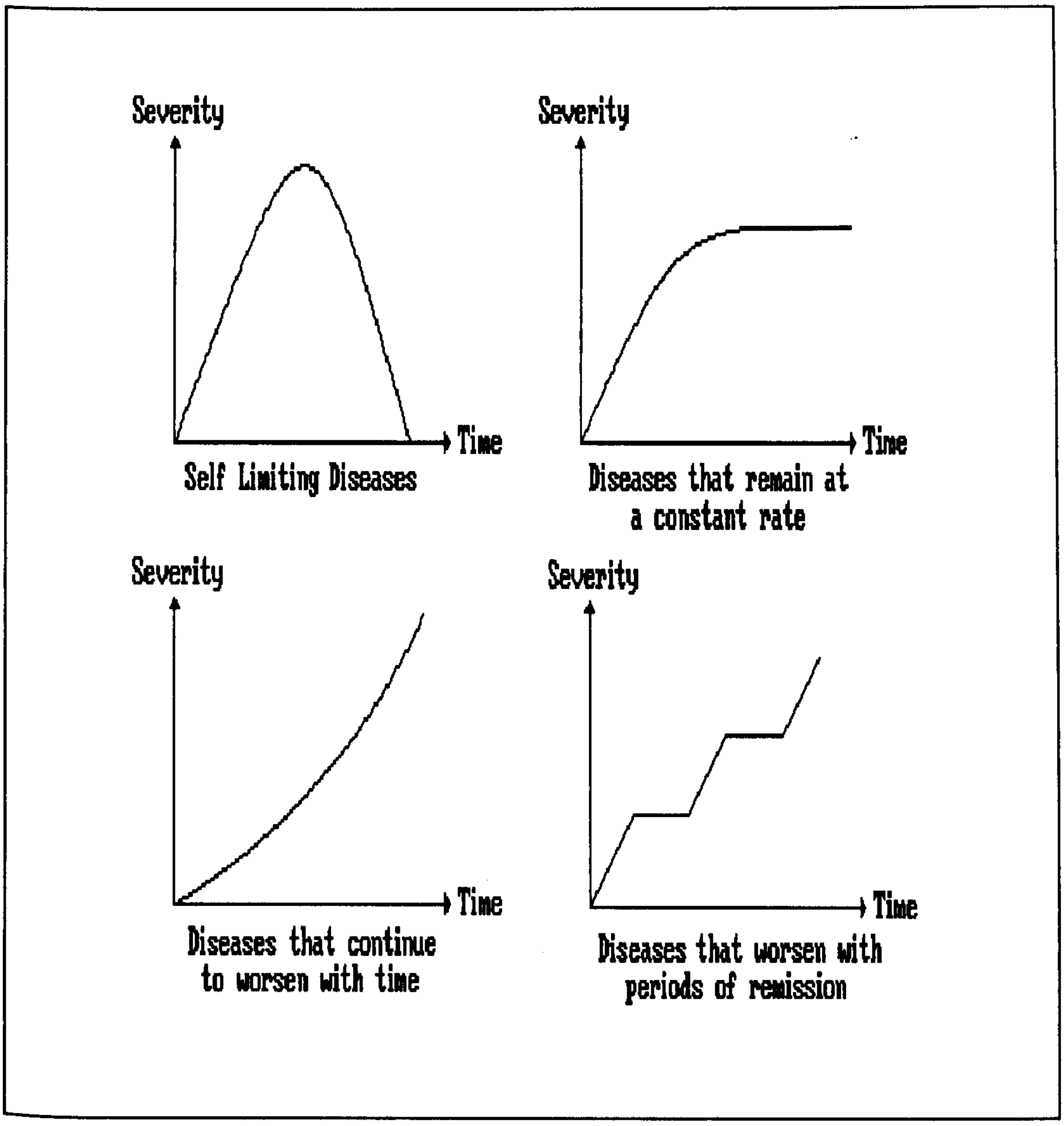


Figure 5.1 - The Four Types of Disease Progression

A screen showing samples of the above graphs, similar to Figure 5.1, is displayed for the person entering the data to select. Once a graph is selected, the screen is redrawn to allow for the entry of graph specific data. These data include the age and sex incidence, disease prevalence, the time the disease reaches its critical level, and the time it takes to end (this last item of data is not applicable to graph types 'c' and 'd' since they represent progressive diseases).

5.1.1.2 The Symptoms and Clinical Signs Entry Procedure

This procedure consists of three sub-procedures; the first one being responsible for maintaining the main screen while the other two are responsible for building sentences to describe the symptoms and clinical signs that are associated with the particular disease.

The main screen has been designed in the form of a spreadsheet or table with two columns; one for the symptoms, and the other for the clinical signs. Each cell in this "spreadsheet" can be viewed or deleted but not edited for consistency reasons.

The symptoms and clinical signs sentence building screens are fairly similar in their operation. Both screens use a system of hierarchical menus to allow the person entering the medical data to construct text strings which are descriptive and varied linguistically, but yet related to underlying codes. The codes are used in place of normal text for minimizing the size of the medical database. The codes and their text equivalent which represent a symbol table are stored in two files; one for the symptoms and the other for the clinical signs. The symptoms and clinical signs are saved in two Prolog lists. A third list is also used for the storage of symptoms that are identified by the General Practitioner to be the main presenting symptoms of the particular disease and therefore wishes to be displayed to the student at the beginning of the case during a tutorial session.

The current version of the system is somewhat limited since it only deals with diseases or medical conditions on a one-off basis. That is, a patient is affected by one disease or medical condition which disappears when treated. This, however, will be rectified in the next version of the system where it will be extended to cater for the entry of symptoms and/or clinical signs for each stage of the disease's life-span, and takes into account other factors such as past medical history, past and present social and economic states, etc.

5.1.1.3 The Laboratory Test Results Entry Procedure

This procedure enables the person entering the medical data to enter the results of all the laboratory tests required (if any) for the diagnosis of a particular disease. The default values for all the laboratory tests (including the times taken to obtain such results) available for request by any General Practitioner are stored in a special file as Prolog facts. These values are displayed on the screen for the medical person entering/editing the disease to modify if needed. As with the symptoms and clinical signs, a symbol table is used to store the names of all the laboratory tests used in the system.

This procedure will also be extended to allow for the entry of laboratory tests that are appropriate to the different stages of the disease's life-span.

5.1.1.4 The Management Plan Entry Procedure

A management plan is a set of actions instigated by the student for the purpose of treating the particular case. There are three types of actions: (1) Curative, (2) Palliative, and (3) Diagnostic. The three types are represented in *P.C.Tutor* by a full set of alternatives or options structured in a hierarchy of menus.

The main options in the management section of the Editor and *P.C.Tutor* include Therapeutic and Diagnostic Interventions, Clinical and Administrative Procedures, Referral to Other Agencies, and Future Reviews. Each of these options is sub-divided into a number of sub-level options.

Once an option is selected, a fill-in screen is then displayed for the entry of all the actions that are related to the particular disease according to the four lists mentioned earlier in this thesis (see section 4.2.1.2 point 5). All actions are entered by selecting them from a set of menus. In addition to the management actions, the person entering the data is given the choice to type in comments to be associated with the particular action for the tutor to display during the tutorial session.

A symbol table is used, in a similar manner to the other data in the system, to store all the management actions text strings.

One of the important features of the management actions or interventions lists is the use of the standard British National Formulary drugs naming system rather than specific drug names, e.g. the drugs Paracetamol and Aspirin are replaced by Simple Analgesics. The advantage of this method is to eliminate student confusion regarding available durgs and their dosages.

As in the previous sections, this procedure will also be expanded to allow for different management plans to be associated with the different stages of a disease's life-span.

5.2 THE MEDICAL TUTORING SYSTEM*

It is significant at this stage to distinguish between a teaching system and a tutoring system since the outcome of both our projects is a tutoring system for use in addition to lectures and readings. B. Woolf [1984] points out that:

> "A teaching system presents new data to a student or provides an environment within which a student can experiment with his or her information, testing and exploring his or her reasoning. While a tutoring system is a specialized teaching system that engages a student in a linguistic exchange for the purpose of clarifying a

---

* Extracts from this section and subsequent sections have been published in the proceedings of the following conferences: (1) Medical Informatics International Conference on Computers in Clinical Medicine, September 1988, Nottingham, U.K. {co-authored by J.Poyser, L.V.Atkinson, J.J.McGregor and M.E.Franklin} (2) Computers in Medical Education, December 1988, Cardiff, U.K., {co-authored by J.Poyser, J.J.McGregor and M.E.Franklin} and (3) The CAL89 Symposium, April 1989, Guildford, U.K., {co-authored by J.Poyser, J.J.McGregor and M.E.Franklin}.

body of knowledge to which the student has already been exposed, e.g. through lectures or readings".

The objective of any intelligent tutor is to enable the computer to understand both the concept to be taught and how the student can learn that concept.

In any General Practice (or even hospital-based) tutorial, the student is given some information about a patient and is expected to obtain more information about the particular case (e.g. through directed questions, observations, or examinations), as he or she deems necessary, to draw conclusions about the patient. The purpose of such a tutorial is to use the context of an actual disease to make the student aware of gaps or inconsistencies in his or her knowledge, and to correct these deficiencies. It is assumed, at this stage, that the student has the technical background necessary to solve these cases.

In the following sections I will be examining in detail the implementation issues used in the CAL system (e.g. patient simulation, generating the tutor's hypothesis list, etc.), and in the intelligent tutor (e.g. student-tutor dialogue management, student modelling and assessment, etc.).

5.2.1 System Implementation

As mentioned earlier in the thesis, the CAL system is the basis for the intelligent tutor. Its function is to simulate a General Practice environment to a student by providing him or her with a presenting problem to manage.

The complete medical tutoring system (i.e. the CAL system and the intelligent tutor) comprises the following stages:

1. Patient generation.
2. The selection of the main disease.
3. The formulation of a differential diagnosis.

4. The main tutoring environment.

Figure 5.2 shows the processes and files involved in the first three stages above. While the model shown in Figure 3.8 represents the processes invoked by the main tutoring environment.

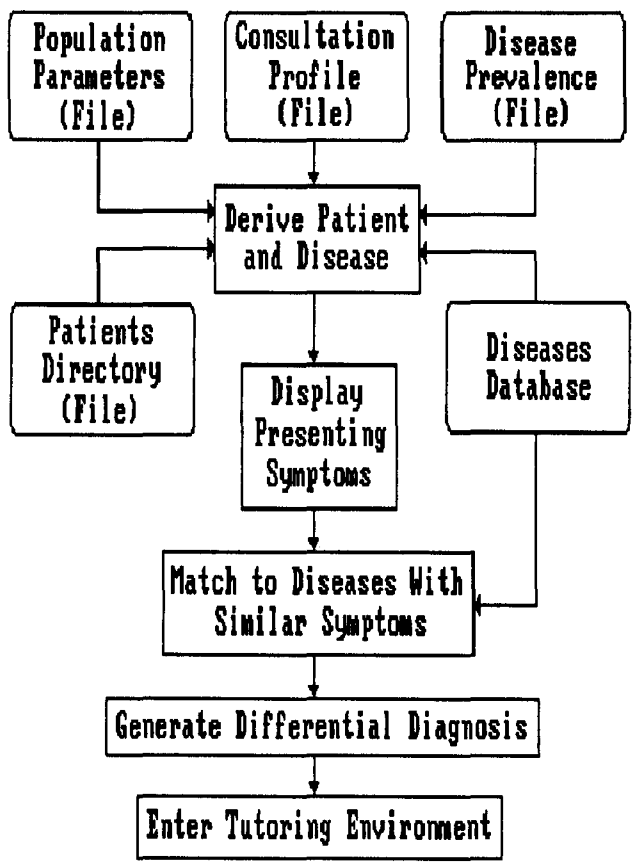


Figure 5.2 - The selection of Patient and Main Problem, and the Generation of the Tutor's Differential Diagnosis

A full data flow diagram of all the files used in *P.C.Tutor* and main stages or processes is shown in Figure 5.3. Stages 1 to 3 above are shown as process 1 in Figure 5.3. While stage 4 is shown as process 2.

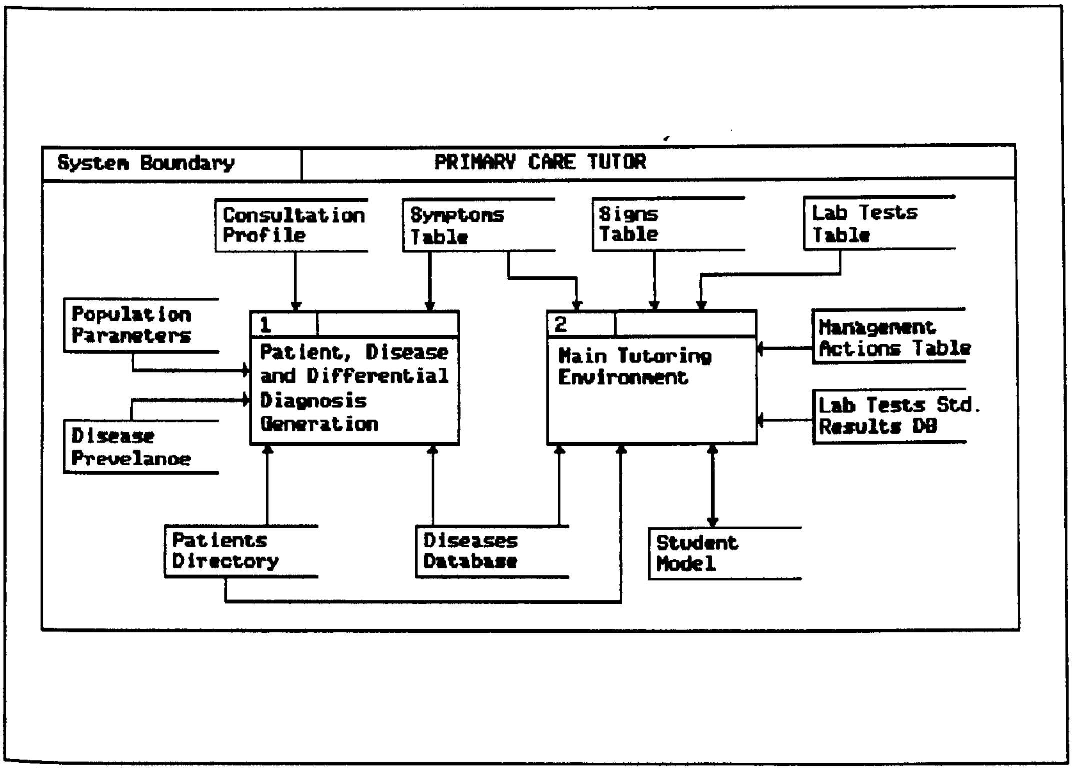


Figure 5.3 - A Data Flow Diagram showing the main files and processes in the P.C.Tutor System

The CAL system starts by generating a patient description which includes the patient's age, sex, and other randomly generated patient details (for the purpose of presentation) such as the patient's name and address. The program takes into consideration the size of the male and female population in a particular community, and the number, age and sex of people visiting the surgery at any time.

The system thereafter selects a disease from the medical database. This selection is based on the generated patient's age and sex, the age incidence, sex incidence, and prevalence of the particular disease.

This greatly helps to provide the student with as near as possible to real-life General Practice work in any teacher selected environment (e.g. Urban, Rural, Third World country, etc.).

Once the patient and disease are selected, the tutor displays a number of presenting symptoms for the student to consider. The tutor then begins assembling a list of diseases which have common presenting symptoms with the selected disease. This list, identified as the tutor's *differential diagnosis*, is used by the tutor to guide the student when requesting assistance, to remind the student of any missed out diseases in his or her hypothesis at the end of the case, and to reject any incorrect diseases mentioned by the student in his or her hypothesis. Incorrect diseases are rejected on entry for the reasons that working on incorrect diseases is time consuming and not relevant to student teaching.

The accuracy of the tutor's initial differential diagnosis depends on the following factors:

a. The number of diseases in the medical database - by having more diseases, there is a possibility that more diseases may share similar case data.

b. The presenting symptoms of each of the diseases in the medical database.

c. The way each of the presenting symptoms is specified - it is very important for the person entering the medical data, when entering the presenting symptoms, to think about the diseases that will be generated by the tutor in its initial differential diagnosis.

d. The disease prevalence and population parameters data.

The tutor's differential is generated by cross referencing the presenting symptoms of the tutor's main disease and the presenting symptoms of the other diseases in the medical database. This process is based on special codes used to identify the positions in the body where the presenting symptoms are noticed. This list is then split into two groups (Primary and Secondary) according to the number of presenting symptoms that were matched with the tutor's selected disease.

The outcome of the tutor's differential generation process is a small database which represents a subset of the complete medical database. This database is used throughout the session for supervising the student-tutor interaction.

Once the tutor's differential diagnosis is made, the student is given the choice of entering his or her initial hypothesis, or acquiring more case data. Having entered his or her reply accordingly, the student enters the main environment.

Throughout the session, the tutor works concurrently with the student on the selected case. In acting this way, the tutor can detect the elimination of diseases, understand to a certain extent the student's problem-solving strategy and so on. When diseases are eliminated, the tutor's differential is refined accordingly. This differential symbolizes what the student should be thinking given the known case data.

The CAL system is a knowledge retrieval oriented system by Woolf's categorisation [Woolf *et al.*, 1984]. This is in contrast to the intelligent tutor which uses some AI techniques, such as the application of tutorial-rules (T-rules) and a model of the student's understanding, as seen by the system, to interact with the student throughout the tutorial session.

When using the *P.C.Tutor*, the student is asked to supply his or her initial hypothesis (i.e. the list of suspected diseases) for the particular case as early as possible. This hypothesis is used by the tutor to update the student model and tailor its responses to the particular student accordingly. In addition to entering the initial hypothesis, the student is also asked to update this hypothesis every time the existence or non-existence of disease(s) is confirmed.

During a tutorial, the tutor concentrates on two areas: (1) the student's differential diagnosis of the case as stated in his or her hypothesis, and (2) the student's management plan for the particular case.

The student's differential (or hypothesis) is in fact a provisional diagnosis. This is refined by the information collected from the patient, and by the results of some clinical examinations. The student is questioned, during or at the end of the case and depending on his or her model, on some of the modifications or refinements that are made to the hypothesis.

The student's management plan of the case is compared by the tutor with the management plan stated by the expert doctor. Any deviations made by the student in his or her plan are stated by the tutor through question and explanation procedures which are generated by some of the T-rules. The tutor is not strict on following the expert's plan to the letter. It allows the student to follow any plan providing that he or she includes the essential components of the management plans of all the diseases in his or her differential diagnosis. In otherwords, the student is allowed to formulate a management plan without having a clearcut view on the diagnosis of the particular case. Nevertheless, the student is not allowed to follow any incorrect route, which may have serious outcomes on the patient's health, during the management of the case. Since we are aiming to simulate a real-life General Practice environment, it is unethical to follow up any serious action(s).

To enable the student to acquire better understanding of the problem in hand, the tutor acts as an observer that only intervenes when the student has missed or overlooked an important or vital operation/action that may have some serious outcomes on the health of the simulated patient, and also when the student model indicates clear uncertainty in the student's knowledge. These interruptions occur as a result of the firing of one or more T-rules. The tutor will, at the end of each of the areas mentioned in section 4.2.4 and at the end of the tutorial session, provide the student with a critical evaluation of the student's handling of the problem. The evaluation is displayed on screen with the option of producing a hardcopy for the student to keep for later reference.

Overall, the computer plays the role of a lecturer in General Practice working with the student on a given case. The symptoms of any case are presented to the student in simple English so as to simulate a patient-to-doctor dialogue. The student-lecturer interaction is realistic, and it allows the student to focus on understanding the patient's problem, rather than the important but secondary issues of communicating with the patient directly. As indicated earlier, such interpersonal skills are unlikely to be acquired from a computer.

The tutor aims that the student should have a clear understanding of the correct diagnosis and management plan of any case presented to him or her during the tutorial sessions.

The main components in the tutoring strategy adopted in *P.C.Tutor* (see Figure 3.8) are discussed later in this chapter.

The following sections provide full details of the components of the tutoring rules used in *P.C.Tutor.*

5.2.1.1 Tutorial Rules

Two categories of tutorial rules are used in *P.C.Tutor*:

1. Rules specific to the medical domain

These include Common Sense and Clinical Examinations rules. Common sense rules are applied to the student's management plan of a particular case. Examples of such rules include referring a male patient to a Gynaecologist, requesting a semen analysis test for a female patient, and so on. Common sense knowledge has been explicitly specified in the source code. There are approximately 25 such rules used in the present version of *P.C.Tutor*. Figure 5.4 shows an example of a common sense rule (see Appendix C for the full list of the medical domain T-rules).

Rule Format:

```
sense(Age, Sex, Management_Action):-<condition>.
```

Prolog Format of an Example Rule:

```
(Age, Sex, sign_contraceptive_advice_form) if
                    Sex = female and
                    Age > = 35.
```

Figure 5.4 - Example of a Common Sense Rule

Clinical examination rules, on the other hand, are applied to the student's fact finding process (Figure 5.5). There are exactly 46 such rules currently implemented in *P.C.Tutor*. Their purpose is to prevent the student from trying out all the clinical examinations in a particular session. All the examinations are linked by these rules to the primary sites of the symptoms and the disease groups.

English Format of an Example Rule:

```
IF
        group is Diseases of the Blood & Blood-Forming Organs, and
        the requested examination is a member of the following list:
                General Condition, or
                URT, or
                Cardiovascular System, or
                Reticulo-Endothelial System, or
                Abdomen, or
                Chest & Thorax
THEN
        report the results of the examination
ELSE
        display "inappropriate examination" error message, and
        amend student level
ENDIF
```

Figure 5.5 - Example of an Examinations Rule

Common sense rules have no effect on the student model. While clinical examination rules result in lowering the student model if inappropriate examinations are made by the student. In addition, any rule from the above two types fired in response to a student action results in a reminder being added to the tutorial log.

Common sense and examination rules can be seen as expert rules in this tutoring system.

2. General Problem solving rules

Such rules are concerned with Disease Elimination and Patient Management. These rules (Figure 5.6) are considered as general rather than specific to General Practice work since they all apply to other problem-solving domains of similar model to that shown in Figure 1.2.

Disease elimination is a process of deduction where the existence of a set of data either proves or disproves the possibility of a problem (medical or otherwise). Patient management, on the other hand, is a process of treating a problem (see Appendix C for the list of problem-solving T-rules). The application of the tutoring system to other domains is discussed in Chapter 7.

English Format of an Example Rule:

IF
    the results of a laboratory test are abnormal
THEN
    mark as *deleted* all the diseases in the tutor's
    differential where the results of this test are normal
ENDIF

Figure 5.6 - Example of a General Problem Solving Rule. The elimination of a disease through the findings of an abnormal laboratory test results.

5.2.1.2 Tutorial Rules Implementation and Choice of Language

The T-rules in *P.C.Tutor* were explicitly defined in the source code. That is, any changes require the re-compilation of the appropriate module(s) in the program. The rationale for the "hardwiring" of T-rules in the source code lies in two areas:

(1) Hardware

As pointed out in Chapter 1, the aim of the CTISS project is for the CAL system to be distributed to medical schools throughout the country. To do so, and on the basis of what is currently available from hardware in such schools, a decision was made by our team to keep the hardware requirements to its basic level. The outcome of this decision is the restriction of the size of memory available for running the system (in our case 640K of RAM that is accessible by MSDOS^{TM}).

The usual method adopted by most ITSs' developers is to write an interpreter for interpreting the T-rules used in their systems. This has the advantage of easily maintained T-rules. However, there are a number of disadvantages in having such an interpreter. These can be summarized as follows:

1. Slower system response time.

2. Extra development time is required for writing and testing the interpreter.

3. Valuable memory space is taken up by the interpreter.

For the above reasons, all our T-rules were explicitly defined in the source code of *P.C.Tutor*.

The problem with hardware is very common when developing complex systems of this kind. In fact, some systems failed to be used outside their research centres mainly because they were originally developed on large AI research workstations.

(2) Software

To be more precise, the choice of programming language. I stress the use of a programming language (e.g. Pascal, Prolog, etc.) in a project like this rather than a CAL authoring language. The rationale for this can be clearly seen in table 5.1.

Authoring Languages	Programming Languages
Expensive	Very Cheap
Poor Control Structure	Good Control Structure
Can not be tailored	Tailored to any application
Poor Documentation	Good Documentation
Slow in Operation	Fast
Requires Runtime Programs for Distribution purposes	Stand-alone Programs
Not Favoured by Software Publishers	Favoured by Publishers

Table 5.1 - Comparison of Authoring Languages and General Purpose Programming Languages

The points summarized in table 5.1 shows the majority views of systems designers and programmers. Furthermore, for the purpose of rapid prototyping, AI languages such as LISP and Prolog are the most appropriate due to the following reasons:

1. AI language environments are interactive.

2. Memory management is handled automatically. Memory allocation is performed automatically by the environment when needed.

3. Type checking is usually automatic.

4. Tracing and debugging facilities are generally more powerful and easier than in some of the imperative programming languages.

As mentioned earlier in this chapter, the Turbo Prolog V2.0 language was used in the development of the complete medical system. This particular implementation of the Prolog language and its programming environment was found to be suitable for the rapid prototyping of the medical system. In addition, Prolog was chosen, as the programming language to use, due to the experience of most of our team members in this language.

5.2.2 Dialogue Management

An important aspect of ITSs is Dialogue Management. For a system to be interesting and effective in getting the subject knowledge to the student requires a clear understanding of the student knowledge. Dialogue management is normally associated with the student model (see section 5.2.3 for full discussion on the implementation of the student model in *P.C.Tutor*).

Although natural language processing has not been built into the system, the tutor has a number of features that enable it to converse with a knowledgeable student in a different way than with a confused student.

In the sections that follow, I will provide in some detail the methods used to achieve this aim.

5.2.2.1 Coping with Incorrect Case Data Requests

If a student requests information we should assume that he or she has a good reason for asking. When we say that a request is incorrect, we mean that an expert General Practitioner does not need the requested information at this time, and may in fact never need it. The possible reasons for requests considered as incorrect are listed below.

1. The request is *premature*: An example of this is a request for a laboratory test, which usually requires time to be processed and might inconvenience the patient, prior to finding out about simpler case data (e.g. symptoms, clinical signs, etc.).

2. The request is *inappropriate* for a General Practitioner: An expert General Practitioner never asks for this information.

3. The request is *relevant, but not useful*: Given the order in which the student has acquired information, he or she does not need to know the answer as it can be evaluated or concluded on the basis of what is already known. This is relevant to all requests for clinical examinations and laboratory tests. In reality, there are many different "correct" queries and sequences of information gathering. A query can be evaluated only if the evaluation takes into account the total context in which the query was made. The selection of strategy by a Practitioner in his or her collection of information about a particular case is critical, since it is neither feasible nor desirable to do all conceivable clinical examination, or to carry out all available laboratory tests. Because of circumstances of available time, possible inconvenience or hazard to the patient, and financial cost, the GP is forced to select only the most appropriate subset of information

to collect. Furthermore, this selection process is not a static decision, but a dynamic activity in which each new item of information modifies the current differential and therefore influences the judgment as to what additional information would be most useful.

The tutor's belief that the student knows or has some idea on the problem in hand is lowered for each of the above reasons (see section 5.2.3.2 for full discussion on credit assignments). Based on the current model, the student is told about the tutor's opinion of the invalidity of his or her request and why it is so.

5.2.2.2 Providing Help

A student can request help at any time during a tutorial session. There are two types of help available in *P.C.Tutor*: *General help*, and *Specific help*. Any request for Specific help results in lowering the student model by a certain pre-defined value (see section 5.2.3.2).

General help

This type of help is used to familiarize the user with all the functions in the *P.C.Tutor*. The help information is stored as text files which are retrieved accordingly.

Specific help

At any time during the tutorial session, the student can request help that is specific to the currently active stage. One example is when the student wishes to make an examination but not completely sure on which one(s) to make. A request for help at that stage will force the tutor to list the appropriate examinations that can be made in the particular case. Some of these examinations may produce normal results. The level of detail given to the student is dependent on the student model.

Specific help is also available for a student entering a hypothesis. At this stage the tutor's strategy is to give the student a maximum of two diseases from its initial differential diagnosis for him or her to consider. These diseases will then be considered as the student's hypothesis. The student should at that stage use his or her process of deduction to find out about the correct disease.

As a special case when providing help during hypothesis entry, diseases that have been previously diagnosed and/or used by the student and which are part of the tutor's initial differential diagnosis, are displayed to the student for analysis. The rest of diseases are given to the student when the next request for this kind of help is made. The main reason for not displaying all the diseases in the tutor's differential is not to give the student too many clues about the particular case.

5.2.2.3 Responding to a Student Hypothesis

At any point in the tutorial session the student can state his or her hypothesis for the problem under consideration. The strategy of the "hypothesis response" T-rules in most CAL systems is to quiz the student about evidence that he or she may probably know or may have forgotten. When the student model is clearcut (strong belief that the student knows or does not know a rule), the tutor accepts the hypothesis without quizzing. The evidence usually consists of all the case data that were displayed to the student prior to entering the hypothesis.

Quizzing a student on his or her hypothesis regularly may have negative outcomes on the effectiveness of the teaching system. The student in this situation is usually guided by the tutor through the implied reasons that instigated the quizzing process in the first place.

To avoid this in *P.C.Tutor*, the student is quizzed only when entering the wrong hypothesis.

The quizzing process is made using multiple choice questions, where any incorrect and missing evidence is explained by the tutor. The evidence is classified into many groups, partly on the basis of the logic of the model, and partly on the basis of distinctions that will be useful for tutoring. These groups are:

1. *Correct hypothesis values* (i.e. the student has specified the same diseases as in the tutor's differential diagnosis).

2. *Wrong values* (i.e. the student has entered diseases that were not considered by the tutor, either because they were eliminated during the generation of the tutor's differential diagnosis, or they have not been entered in the medical database).

3. *Missing values* (i.e. the student has missed out one or more diseases, part of the tutor's differential diagnosis, from his or her hypothesis).

As a special case, a student can (if he or she wishes to) enter his or her initial hypothesis on the basis of the patient record and the presenting symptoms of the problem in hand (i.e. at the start of the case). In this case, the student will be quizzed only when the diseases entered are not in the tutor's differential diagnosis. The reason for this is to ensure that the student is always working on tutor recognized diseases only.

The evidence for any correctly stated disease(s) will not be given at this stage of the tutorial session.

Finally, as mentioned earlier in Chapter 3, case diagnosis is not central in General Practice work. However, in *P.C.Tutor*, the process of disease elimination or hypothesis refinement used by the student during the tutorial session is considered as a diagnostic procedure.

Premature elimination of a disease (i.e. given the already known information, the tutor can not arrive at such a conclusion) forces the tutor to, depending on the student model, either request the student to justify his or her action, or reject the action for insufficient evidence. If the student was required to justify his or her action, *P.C.Tutor* will inform him or her of the reasons for considering the action as premature.

Question Formats

The question formats, some of which are based on GUIDON's question formats [Clancey, 1987a], are as follows (The dollar sign $ is used to separate the premise and action parts):

(a) Premise Format: Multiple Choice
Action Format : Wrong Hypothesis Hypothesis Refinement
"What facts about this case $ tell you that <Actual Conclusion> is present: Select from 1...N"

This format is used to request the student to support his or her hypothesis by selecting the relevant facts. Valid selection is accepted but overall the student action is rejected on the basis of other known case data.

(b) Premise Format: Multiple Choice
Action Format : Premature Hypothesis Refinement
"What facts about this case $ tell you that <Actual Conclusion> has been eliminated: Select from 1...N"

This format is used to request the student to support his or her hypothesis refinement by selecting the relevant facts. Valid selection is accepted but overall the student action is rejected on the basis of incomplete case data.

(c) Premise Format: Actual Value or Wrong Value
Action Format : Right or Wrong Hypothesis
"Given <premise facts> $ one would conclude <Actual Conclusion>?"

Here, the student is requested to confirm the correctness or incorrectness of the given statement.

The two types of question formats are fired according to student actions and the student model.

5.2.2.4 Providing for and Coping with Student Initiative

An essential part of tutorial dialogue management is allowing the student to express him- or herself. To construct a mixed-initiative program, provision must be made for every potential kind of initiative that the student will be allowed to take. We might summarize this by saying that we must allow the student to specify what he or she knows, wants to know more about, and wants to ignore.

P.C.Tutor provides for student initiative by giving the student a wide variety of options (input commands). The options available to a student are summarized in table 5.2. As described in the following sections, each option or group of options generally has a discourse procedure associated with it. The program uses single-word options and hierarchical menus as these are more convenient, in terms of simplicity and speed, for use by the medical students. Options are activated using either specially defined function keys or special characters associated with each option.

Option Type	Summary of Options
Getting Case Data	SYMPTOMS
	EXAMINATIONS
	TESTS
Conveying what the student know	HYPOTHESIS
Case Control	MANAGEMENT
Requesting Assistance	HELP
System Options	VIEW
	REVIEW SYMPTOMS
	REVIEW EXAMINATIONS
	REVIEW LAB-TESTS
	PATIENT RECORD
	PRINT LOG
	NOTE PAD

Table 5.2 - Student Initiative Options

The discourse procedures used by the intelligent tutor are dependent on the student model and some are activated after each student action. Each procedure consists of a number of production rules. The rules, implemented in the tutor, are applicable to all the diseases in the medical database.

The sections that follow are organized according to the option groups in table 5.2.

Getting Case Data

The three options in this section enable the student to reach a final diagnosis or an optimal sub-set of diseases that share the same information of the case in hand. These options are used to retrieve case data from the main database. Two procedures are used during the fact finding stage of the tutorial session. The first procedure is called after each request for case data. Some of the rules in this procedure are based on the process of disease elimination or streamlined diagnosis. The other procedure is used to explore the student's knowledge about diseases considered in the tutor's differential diagnosis.

(i) SYMPTOMS

In addition to the presenting symptoms which are displayed at the beginning of the case, the student can request more symptoms. The program gives the student the choice of two types of requests: *Undirected* and *Directed* requests. An undirected request is in the form of "Next Important Presenting Symptom" which is more common in the General Practice environment. Here, the program responds with sentences that simulate a patient-to-doctor conversation (e.g. Dull headache for the last 36 hours). Undirected requests are applicable to presenting symptoms that were not given at the start of the case.

Directed requests, on the other hand, are made by selecting from a number of menus which represent the main symptom areas (e.g. Altered Sensorium, Self-Perceived Signs, etc.)

and the symptoms in general (e.g. Aches and Pains, Vision problems, Shivering, Skin rash, etc.). Directed requests apply to all symptoms i.e. presenting and non-presenting.

All directed requests made by the student are checked for the existence of such symptoms in the main tutor's selected disease. Symptoms not part of this disease are given as "Normal".

(ii) EXAMINATIONS

The word 'Examinations' includes the clinical signs observed and examinations made by the General Practitioner at his or her surgery.

An examination can be made by selecting the clinical area from the main examinations menu. By dividing the examinations into clinical areas, only area-related-data are presented to the student, rather than the tutor disgorging a plethora of signs at a single request.

Once an area has been selected, another menu consisting of action options (e.g. Inspect, Palpate, Check heart beat, etc.) is displayed according to the selected area (i.e. each area has its own set of actions).

The tutor responds by displaying all the signs that are appropriate to the particular examination (if any were found). If, however, no abnormal signs were found to exist, the default value for the examination is given (usually the word "Normal").

(iii) LABORATORY TESTS

These include the tests made by the General Practitioner at his or her surgery. Results of tests which have been found negative are given to the student from the file of standard tests values. Laboratory tests with numerical values are stored as a range of two values (a minimum and a maximum value). When a student requests a test whose result is a numerical value the tutor randomly generates a value between the two limits. The two limits which represent the reference range are also displayed to the student for him or her to interpret whether or not the particular test is normal or abnormal. During its generation of random values for representing abnormal results, the tutor makes sure that the selected value is outside the default range for that test.

Laboratory tests which cannot be performed at the surgery are considered part of the management plan for the particular case. In such a case, the student can request any number of tests, which he or she considers to be necessary for the particular case. The results of such tests will be given at the next tutorial session. The main reason for following this sequence of operations is to provide some real-life simulation of the work of General Practitioners. At the stage when the test results are given to the student, the tutor will, depending on the student model, question the student on the relevancy of such tests.

Conveying What The Student Knows: HYPOTHESIS

As indicated earlier, the student is asked to enter his or her initial hypothesis as early in the tutorial session as possible. Having entered the initial hypothesis, the student can, at any stage from that point onwards, update his or her hypothesis by either deleting incorrect or adding new diseases to the current list. A procedure is used every time the student is in the process of updating the current hypothesis and also during the entry of the initial hypothesis. This procedure is used to update the student model and respond to student hypothesis accordingly.

The student is also questioned at the end of the session on any missing diseases that were considered by the tutor in its differential diagnosis. This acts as a reminder for the student to consider when faced with the same problem in the future.

Case Control: MANAGEMENT

Setting up and implementing a management plan is one of the two important processes in the tutorial system (the other process is the formulation and streamlining of differential diagnosis). The student is given appropriate feedback on his or her management plan. In case when the student plan is totally wrong and in addition to the feedback, the correct plan for the case is given to the student. An incorrect plan is considered by the program to have or possibly may lead to negative results for the particular case.

Formulating and implementing a management plan involves using a procedure which determines all tutor responses to the particular student plan.

The current version of the tutor allows the student to enter one management plan for each case. This will later be changed to allow the student to enter more than one management plan if deemed necessary when the system starts to simulate the changing state of the patient as a result of any laboratory test, previous interventions, etc.

Requesting Assistance: HELP

A student can request help at any time during the tutorial session. The tutor aims to provide different help information for different students. Help is given according to the currently used discourse procedure and the student level of understanding (i.e. the student model). In addition to the help related to the working of the particular case, the student can also get help on using the system (i.e. General Help).

System Options: VIEW, PATIENT RECORD, REVIEW <case-data-type>, PRINT LOG, and NOTE PAD

The VIEW option allows the student to review the complete current tutorial log. The log will include the presenting symptoms, all the results of the requests for case data, all hypothesis updates, the student's management plan, and tutor feedback.

The PATIENT RECORD option enables the student to review the personal information about the particular patient. The two important values in the patient record are the patient age and sex. The third option is the REVIEW <case-data-type> which is used to display the symptoms, examinations or laboratory tests in a separate window to help the student to consider the appropriate data more carefully than using the VIEW option. The fourth option PRINT LOG is used to print the current tutorial log. The last option NOTE PAD offers the student a simple word processor for writing any notes during the session. The NOTE PAD is also useful from a lecturer point of view, in that he or she can get student responses from his or her students on the contents of the medical database. The five options have no discourse procedures associated with them.

5.2.3 The Student Model - Its Implementation and Maintenance

5.2.3.1 Model Implementation

The following sections provide details of the contents of the two components of the student model (see section 4.2.5) and explanations on the method applied in implementing these components.

Background Component

This component is first generated at the beginning of the first session for each student. It is updated accordingly during the following sessions using information from the Current Model. The data for the Background component is stored in a special file which is accessed at the beginning of each session. The information that is used in this component consists of:

a. *The student level* - this is a numeric value of the tutor's estimation of the student problem-solving experience (or understanding).

b. *The used diseases* - this is a list of all the diseases used by the student in his or her differential diagnosis during all the tutorial sessions. The list also includes any diseases given to the student by the tutor when requesting assistance.

c. *The tutor's selected diseases* - this is a list of all the tutor's main diseases that the particular student has come across during the previous sessions.

d. *The overall student level in the case data gathering process.*

e. *The overall student level in the formulation of differential diagnosis* - this number represents the student's competence in formulating his or her hypotheses.

f. *The overall student level in the formulation of management plans.*

g. *The overall student level in hypothesis refinement (disease diagnosis).*

Current Model Component

This component is a temporary component used during the current tutorial session only. In addition to representing the student understanding during the session, it is also used to update the Background component for future activities. This component consists of the following information:

a. *The student's hypothesis* - this is a list of all the diseases considered by the student during a session.

b. *The unknown diseases* - this list consists of all the diseases that have been missed out by the student during a session. A missed out disease is one which is part of the tutor's differential diagnosis with very similar presenting symptoms.

c. *The current student level* - this is a numeric value which represent the tutor's confidence in the student's knowledge in the current case.

The Current Model component is continuously updated from the following sources:

a. *Student hypothesis* - the student is given the choice to update his or her hypothesis at any time during the session.

b. *Requests for case data* - any request for data is checked for its relevance using the student's current hypothesis. Tutor responses are then made accordingly.

c. *Requests for assistance* - any request for assistance is interpreted as missing knowledge in the student's understanding of the particular disease.

d. *Replies to tutor questions* - tutor questions are in the form of multiple choice and true/false questions. Its reactions or responses are based on the student model.

5.2.3.2 Credit Assignment

A system of credit assignments is used by the student model to provide the tutor with some indications about the student knowledge of the case in hand. The student model uses a bipolar scale, shown in Figure 5.7, to represent an estimation of the student knowledge in a particular case.

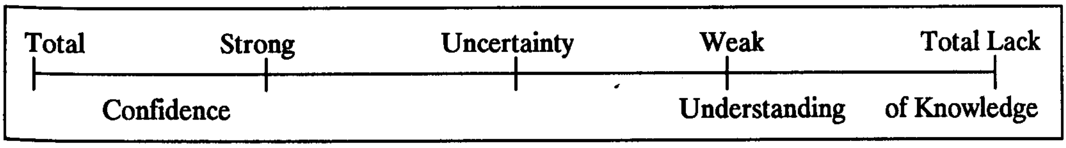


Figure 5.7 - Student Model Scale

The assignment of credits is made according to five main areas of student-tutor interaction given in section 4.2.4. Each one of these areas described below has different weights allocated to it. The allocation of weights are dependent upon the importance of the particular area in the tutoring process. These areas are:

1. Case Data Gathering - This is a difficult area in medicine since one student may need all the data to conclude his or her differential diagnosis, while another may only require a small part of the data for such conclusion. Due to this, no weight is allocated to this area. However, making an incorrect request for a laboratory test results in reducing the student overall mark by a certain amount.

2. Formulation of Differential Diagnosis - This is sub-divided into two sub-areas: Hypothesis and Diagnosis (or disease elimination or hypothesis refinement). The weight for each sub-area is based on the number of primary diseases (i.e. those that have common presenting symptoms and are applicable to the same age and sex values) generated by the tutor. In addition, any primary disease given by the tutor, in response to a request for HELP, is deducted from the Hypothesis overall weight.

3. Management Plan Formulation - The weight related to this area is allocated to three of the four types of actions given in section 4.2.1.2. with the exception to the optional actions list. Optional actions represent a matter of judgment exercised by the student when managing a particular patient. Any missed out important actions or harmful actions made by the student have negative weights allocated to them.

All weights used in this system can be modified by the teacher as necessary. Further, all weights are used internally by the tutor and none is shown to the student. It is also used to explicitly change the context from one area to another.

The purpose of the weights system is to invoke different actions or discourse procedures during the student-tutor interaction.

At the end of a case, the student is given a critical analysis of his or her performance through each of the above areas and, as mentioned earlier, his or her overall performance in the form of a graph.

CHAPTER VI

TESTING RESULTS

6.1 STUDENT OPINIONS OF THE P.C.TUTOR SYSTEM

The first pilot testing trials of *P.C.Tutor* were made through a number of informal sessions involving one user at a time. The aim of these trials was to validate the tutoring strategy adopted by *P.C.Tutor* and the appropriateness of its user interface. For these reasons and the fact that most students taking the General Practice course were on field training during the trials, the majority of students who took part in the trials were either graduates or from the final year of the medical course.

In order for the tutoring system to be objectively evaluated by its end-users, all students were questioned before and after the testing process. The questions asked during both stages were supplied as a two-part questionnaire for the participants to fill in. The questionnaire shown in Figure 6.1 is primarily concerned with the computer science field and in particular the student-tutor interaction.

P.C.Tutor trials were conducted in 12 half-hour sessions. 12 students took part in these trials. Each student was given a set of problems selected on a random basis from the system's 30-disease database. This section, based on the questionnaire in Figure 6.1, outlines students responses and opinions on this system and medical CAL systems in general.

PART I - Pre-Test Questions

1. What is your opinion on the use of computers in medical education? Does the idea of Computer-Aided Learning (CAL) appeal to you?
2. Have you ever used a computer before? If yes, state for what application(s)?
3. Have you any idea on what a CAL system would include?

PART II - Post-Test Questions

1. What is your opinion on the responses and messáges given by the tutor? Do you think that the tutor responds quickly enough to your actions?

2. Do you find the system easy to use?

3. Did you ever feel lost in the tutoring environment itself and did not know what to do next? (e.g. you did not know where you were or did not know how to obtain a certain type of examination, symptom, etc.)

4. Tutor explanations - Are they adequate or can they be improved (say in what way they are adequate or how if they can be improved)?

5. Can you attribute the errors in your answers to:

   a. Lack of medical knowledge
   b. Carelessness
   c. Using the keyboard/Typing
   d. Inadequate instruction
   e. Student confusion

6. Did you ever find yourself stuck on the medical problem and the tutor did not give you any helpful advice?

7. How realistic is the simulation of a General Practice environment?

8. Do you feel that the tutor is guiding you towards the solution? (Please qualify)

9. Do you feel that this tutoring system will be of any benefit to your learning as an undergraduate? How does it fit in with the rest of your work?

10. The following are options under consideration for the future development of this system. Please write your opinion on each of these options:

    a. An electronic reference book (This is NOT a replacement to any textbook). This 'book' will allow you to perform rapid cross referencing of information that is difficult to do manually. It will also give you full details on any facts you wish to know more about.

    b. A differential diagnosis generation module that can provide a differential on any given student supplied parameters (e.g. symptoms, signs, patient age, sex, etc.).

    c. An examination module using Multiple Choice Questions (MCQs).

    d. The addition of more MCQs as part of the tutor discussion of a particular case.

    e. The ability to interrogate the tutor by typing simple English questions or sentences.

    f. The use of video sequences of patients explaining their problems instead of the current text description.

    g. The use of still pictures of injuries, lumps, rashes, X-Rays, ECGs, etc. as a replacement to the current text description.

11.What other additions do you think are necessary for this system to be a comprehensive interactive learning environment?

Medical Content

1. What is your opinion on the presentation of the medical facts?
2. Are there any medical data that you think are missing and should be added?

Figure 6.1 - P.C.Tutor Student Evaluation Questionnaire

The pre-test questions attempted to determine the students' impressions of and any conceptions they might have of computer-assisted learning. While the post-test questions attempted to determine their attitudes to this method of teaching. The following are summaries of the students responses to the questionnaire in Figure 6.1.

6.1.1 Pre-Test Questions

The use of computers in medical education was initially felt by most students as an important educational and training method. Most of the students stressed the point that computer-based teaching/tutoring systems will never act as a substitute for clinical training in wards and clinics. However, they could act as an alternative to tutorial teaching which is acceptable and desirable to most medical students.

All but one of the students who attended the pilot test of the tutoring system had never used a computer before. This point has outlined the need for an orientation session in order that all students get accustomed to using the computer in general and the system in particular.

None of the users had any idea of what a CAL system would include. This is probably due to the lack of CAL systems in secondary schools, colleges of further education, and medical schools. In terms of secondary education, this aspect has started to change over the last seven years mainly due to the introduction of cheap microcomputers (e.g. BBC B, Acorn Electron(TM), Amstrad(TM), etc.) in these establishments.

6.1.2 Post-Test Questions

The fact that nearly all the students had no experience in using a computer was not a factor that concerned them. Apart from the advice which was given to them on the important points in using the keyboard and following the system instructions, they all managed surprisingly well - better than was expected from first time users.

In response to the question regarding where most of the students' errors came from, they all agreed on two points: (1) lack of medical knowledge, and (2) student confusion. Lack of medical knowledge was mainly due to the tutor selecting uncommon diseases (e.g. Hand, Foot and Mouth disease, Bornholm disease, etc.). While student confusion was a result of the tutor forcing them in an indirect way to ask for the right set of symptoms, examinations, and laboratory tests when following up a list of hypotheses. Medical students are taught not to ask, as far as possible, direct questions during a patient interview. Instead, they should allow the patient to fully present his or her problem first. Directed questioning should only start once a differential has been formulated. However, for a computer-based system requiring to build a model of the student level of understanding, directed questions is the only method for achieving this aim.

The students unanimously agreed on the speed of the tutor responses to all their actions during data gathering and patient management. Apart from a few who commented on the need for more detailed tutor responses, the majority of them thought that tutor discussion was adequate. The tutor responses were valid on most occasions and the idea of displaying any missed out facts was considered by many to be very appropriate. They also found that the tutor's explanations regarding any incorrect actions were very helpful. Regarding the need for detailed tutor responses, the amount of detail the tutor can supply is totally constrained by the limited data in the medical database.

With regard to the actual simulation of the General Practice environment, most students thought that the system did not offer a very realistic simulation. Most of them wanted to know more about the patients (e.g. past histories such as medical, social, etc.) particularly when they were managing a complicated case. All students felt the need for visual media to be part of any medical CAL system. The use of visual media helps the student to have a clearer assessment of the patient and his or her problem. The fact that visual media are used in the training and examination of students, is an important design feature in the development of medical CAL systems. The use of visual media in the form of Interactive Video by some medical CAL developers in their "simple" systems has proved to be a successful design recipe.

The students overall opinion of the simulation, taking into consideration its current limitations, was very encouraging. This was also reflected in their response to the question regarding tutor guidance.

Nearly two thirds of the students felt that the system was guiding them to the solution when it rejected those diseases that were not part of its differential diagnosis. However, most felt that, apart from this, the tutor did not give them any clues to the right solution. The other third felt that the system was guiding them but they were unable to justify their opinions. They could not "pin-point" the areas where such guidance was clearly felt. However, they also agreed that the tutor did not tell them whether they were "getting warm" or give them any hints for correctly diagnosing and managing their patients.

When the students were asked for their opinion on some of the suggested future developments for the system, their responses where as follows:

1. *Electronic text-book:* This is a very good idea especially if it contained full references. Providing students with full references is an important system design issue which has been taken up by a number of educational establishments and commercial companies in their development of electronic reference books (e.g. the electronic edition of the Oxford Book of Medicine, Oxford University Press; MEDLINE, etc.).

2. *Differential diagnosis module:* The majority of students felt that such a module would be of benefit to practising General Practitioners rather than to undergraduates.

3. *Multiple choice questions (MCQs) examination module:* Here the students were divided into two camps. Those who thought that an MCQs examination module would be a good practice for them, and those who were not keen on the idea on the basis that a student should learn the whole subject area rather than the MCQ options.

4. *MCQs as part of the tutor discussion:* Some of the students thought that this (picking 1 of 5 options) would be cheating. While others felt that an MCQ with full justification of why an option is correct or incorrect would be a valuable addition to the system.

5. *Natural language entry:* The ability to interrogate the tutor by typing simple English sentences was considered by most students to be a nice addition though not particularly practical. This was basically due to the fact that most of them had never used a computer before and hence had difficulty in using the keyboard.

6. *Video sequences and still pictures:* As mentioned earlier, all students thought having video sequences of patients explaining their problems would be an improvement over the current text-based presentation method. However, the use of still pictures of injuries, lumps, rashes, ECGs, etc. in their opinion was a must in any CAL system.

Finally, in terms of medical content, most students felt that the presentation of medical facts was short and not very explanatory. They also wanted to ask for more information but were not able to do so. The inability to ask the tutor for elaboration of specific points was considered by most as a serious shortcoming in the presentation of medical facts. Apart from this, some of them felt that nothing else was missing from the medical side, while others saw the need for past medical histories and other background information about the patients when treating complicated cases.

6.1.3 Students' Comments

- "The simulation was not realistic because of the lack of visual images."

- "I liked the questions at the end and if these were used in conjunction with a text book it would be a very good diagnostic aid."
- "The presentation of medical facts were short and sweet and not very explanatory. They also lacked the interpersonal communication of a face-to-face encounter."
- "Most students think they have got enough work to do as it is and will not use the system unless it is compulsory. They will regard the system as fun rather than serious learning."
- "The system was nothing like the rest of the work I have done before. I think it would be of benefit to all students and was more interesting than reading a text book."

6.1.4 Concluding Remarks

The students participating in the *P.C.Tutor* trials had to think critically and systematically, to weigh up the value of different actions and were confronted with a series of problems each of which had to be solved. In CAL, personal attitudes and the grey areas of medicine are stressed, perhaps more than in other parts of the undergraduate course.

This was clearly evident in the trials. Most students felt that they were under pressure with the system. They attributed such pressure to trying not to make errors in their clinical judgment when managing cases. It is somewhat surprising that most students felt under pressure since one would expect them to self-pace at an agreeable rate. However, one would expect that with further use of the system they may develop a better attitude to it.

From the students' responses to the post-test questions, the majority of the users were enthusiastic about this form of teaching and were keen to have further such tuition. They also felt that it is an excellent training aid and should be encouraged in all fields of medicine. However, others felt that, given the workload on medical students, medical CAL systems will not be used unless they are compulsory. Their view about such systems is that these systems are simply a method of testing their knowledge in an entertaining way. This is a serious comment in the age of information technology.

Evaluating a tutoring system is a difficult matter since there are no formal methods of evaluation. To fully evaluate this system, a testing period will take place in the first term of the 1989-1990 academic year. In this process, students will be divided into two groups where one group will be given tutorials by this system while the others have normal tutorials. The objective is to show that there is no difference between the two groups.

What conclusions may be drawn from these trials? Computers are increasingly being employed by General Practitioners performing a variety of tasks. This wide variety of experiences with computers (including CAL) should also be offered to students during their training. The undergraduate curriculum is normally concerned with the acquirement of knowledge whereas the work of a doctor is concerned with applying this knowledge to the 'real life' situation. CAL is an attempt to condense this teaching period and to give the student an opportunity of applying his or her knowledge to a 'real life' situation. In addition, it is a convenient education tool which has been shown on numerous occasions in the past to be as educationally effective as tutorial teaching. CAL systems give students the opportunity of testing their knowledge and if this is lacking to learn from their peers. Therefore, contact with lecturers is important and CAL systems may be most important in carrying out repetitive aspects of teaching, leaving the lecturer to concentrate on the student's individual needs.

6.2 PRACTISING GENERAL PRACTITIONERS, HOSPITAL CONSULTANTS, AND OTHER PROFESSIONALS' IMPRESSIONS OF THE P.C.TUTOR SYSTEM

Demonstrating the first version of the system to doctors from two different environments provided some interesting mixed reactions.

From a hospital viewpoint, consultants (some of whom are postgraduate lecturers) have expressed their opinions on a number of points in the system. These are summarized below:

1. The validity of the medical data - Some consultants suggested that the medical data should either be entered by specialist doctors or entered and reviewed by at least two General Practitioners. This is an important point which can also be attributed to personal opinion, experience, and work environment. In other words, different doctors in different environments have varying priorities and representation of medical knowledge. Because the system consists of a database editor, the author can modify the contents of the disease entities without any difficulty. In addition, there were some suggestions regarding the optimum management plans of some diseases. These suggestions, although absolutely valid, are inappropriate to the Primary Care environment due to the limited resources offered to General Practitioners. One example was the action of sending a patient with a Lumbosacral Strain (or low back injury or pain) to physiotherapist. Although correct, sending a patient to a physiotherapist can not be done by a General Practitioner in Sheffield.

2. The need to include other information that can be given to the student in each case. Such information include:

   - Past Medical History
   - Drug History
   - Personal History
   - Social History (Smoking Habits, Alcohol Drinking, Occupation, etc.)
   - Family History
   - Allergies
   - Female related problems

The addition of the above information in any future versions of the tutoring system have already been agreed on.

The consultants' overall assessment of the system was notably positive for the instruction of medical undergraduates.

The system was also demonstrated to a number of General Practitioners working as lecturers in some of the medical schools in the U.K. or involved in the training of undergraduates. Their reactions were extremely favourable and most have expressed their willingness to use it in their departments or surgeries. This reaction is mostly due to the lack of similar systems for use in the General Practice environment as I have shown earlier in section 3.4. Further, when the system was demonstrated at the AUTGP conference at the University of Liverpool in July 1989, two General Practice departments from Oxford and Liverpool universities required a copy of the system for use by their medical undergraduates.

All GPs and hospital consultants who had the opportunity to see and use *P.C.Tutor* were impressed by its user interface. They also verified the validity of the teaching (or more precisely the tutoring) method adopted by the system.

To summarize, the overall impression of all the GPs and consultants, who had hands-on experience of using our system, was the great potential of the system for the teaching of undergraduate medical students aiming to work in both environments. In fact, some doctors suggested the use of a modified version of the tutor for instruction of medical postgraduates.

In addition, the system was demonstrated to people from the nursing and veterinary professions. The nursing profession was represented by nurses and lecturers from one of the nursing schools in the U.K. Apart from agreeing on point (2) above, they felt that a slightly modified *P.C.Tutor* (in the nursing profession, the main emphasis is on the social element in each patient problem) should also be offered to nursing schools due to the similar problem-solving method used in the training of nurses.

The veterinary profession was represented by veterinary surgeons from Bristol University, who in conjunction with the University of Leeds have developed the VIPA system* and are also responsible for running the Computers in Teaching Initiative Centre for Medicine. The system was demonstrated to them on two occasions. On both occasions the system was positively received for its user interface and teaching strategy. In this profession, the model adopted by *P.C.Tutor* closely resembles the consultation model adopted by all veterinary surgeons. The overall emphasis in the veterinary environment is on patient management rather than exact disease diagnosis, an emphasis which is shared with the General Practice environment.

* VIPA is an interactive video-based system. It consists of two systems: (1) a tutoring system based on MCQs and simple question and answer techniques that are based on still pathology slides, and (2) an authoring system for building the tutorials. The system is based on the BBC MASTER microcomputer with a 10MB hard disk and a Philips video-disk system. Both systems are written in BBC BASIC(TM).

CHAPTER VII

CONCLUSIONS

The conclusions of this research are presented in this chapter. Some extensions to the system and future research proposals are also described in this chapter.

7.1 COMPUTERS IN MEDICAL EDUCATION

In this thesis, I attempted to explore the subject of Computers in Medical Education with particular emphasis on the General Practice environment.

Computer-based medical education aims to explore innovative, efficient and cost-effective means of providing instruction in the medical domain. It attempts to do this by adapting and expanding the technology of computer-aided learning (CAL) to develop educational skills beyond the lecture and demonstration that would otherwise be costly or impossible to provide.

In chapter 2 we examined the many forms of teaching strategies or paradigms used in medical CAL systems. These ranged from the simple drill-and-practice method to the complex simulations strategies. The purpose of Drill-and-practice programs is to assist in the memorization of factual material. The paradigm used by these programs is one of presenting a large number of simple, straightforward questions dealing with unambiguous factual information such as term definitions. More complex types of system are testing systems which are designed for evaluation of knowledge and competence. Typical uses include evaluation of student performance, practising for examinations, and self-assessment to determine level of knowledge and needs for further learning. These systems generally consist of large data banks of multiple-choice questions categorized according to content and level of difficulty. Other methods of testing include patient management problems (PMPs) which focus on the evaluation of problem-solving skills.

Further along the complexity spectrum are tutorial systems - direct descendants of the programmed learning approach to teaching. Tutorial systems ranges from the simple "page-turners" that do no more than present textbook material in another format, to the relatively sophisticated systems that incorporate complex branching to guide the learning process and incorporate a variety of features, including slides, audio material, or VideoDisks to simulate accurately real situations and provide tailored feedback on performance. The majority of medical CAL material available today falls into the category of tutorials.

The most sophisticated CAL systems are those based on simulations. Simulations attempt to model real world situations in order to observe behaviour or to practise problem-solving and decision-making. This type of CAL system was the one chosen for the *P.C.Tutor* for the reason that the undergraduate course for General Practice is allocated less time than the hospital-based course, even though about 50% of the undergraduates choose to work in the Primary Care environment at the end of their studies.

7.2 A REVIEW OF THE GOALS

The *P.C.Tutor* system was developed in response to the need for tutoring systems that complement the teaching of medical undergraduates of General Practice.

The scale of the *P.C.Tutor* project was small in hardware and software terms yet massive in practical terms. Many CAL systems have been designed in the medical field (see Chapter 3), but few have attempted to encompass the Generalist Doctor (e.g. Case-History model of the Glasgow University system, see Chapter 3). The difficulties of designing CAL systems within specialties or sub-specialties are daunting enough, but General Practice can encompass *any* known clinical entity, for the analysis of undifferentiated problems is the specific task of the generalist. In so far as there is any curriculum to which the tutoring system could work, the original aim included the brief to enable the system to simulate not only the specific disease entity, but also place that clinical situation within the appropriate context for the population which was being simulated.

The intelligent tutoring module of the *P.C.Tutor* was designed to provide a method for modelling student performance and generating responses that accommodates all student levels of understanding.

The design framework developed in this joint project is not final. I stress the point that rapidly prototyping a system of this magnitude in a short time has the consequence of limiting its features. Additional features such as the inclusion of more T-rules, natural language processing capabilities, and many more could be added to the system. However, with regard to the *P.C.Tutor* project, the intent has been to develop a full working prototype programmed (rather than using an authoring language) from scratch, suitable for use by medical undergraduates, in a period of two years.

7.3 RESEARCH OUTCOMES

The tutoring system developed as a result of this research project has been very successful from the academic point of view. The research has focused on two viewpoints; (1) a medical point concerning the teaching of undergraduate students in medical education in general and Primary Care in particular, and (2) a computer science point regarding the application of the system to other domains with similar problem-solving strategies.

7.3.1 The Medical Viewpoint

The underlying problem with all medical instructional systems is their emphasis on disease diagnosis. Such emphasis is valid in the hospital or Secondary Care environment. However, in the Primary Care environment, the main emphasis that is placed by the practitioner during the five to ten minutes patient encounter at the surgery is to bring the patient to health in whatever way that is quickest, cost effective to the patient and the health service, and convenient to the patient. Faced with these objectives, the practitioner has to strike a balance that is appropriate to the patient and to the health service (e.g. sending patients for laboratory tests or to Secondary Care when absolutely necessary). The GP often relies on his or her clinical knowledge and judgment in the interpretation of the usually "unstructured" patient problem. Patients discussing or describing their problems to the GP frequently include some irrelevant or exaggerated information about their cases. The GP at this stage must be able to "weed-out" such information and formulate a list of hypotheses (or more accurately the differential diagnosis) that can be tested and streamlined on the basis of further evidence.

The majority of Medical teaching systems developed in Europe still leave a lot to be desired. In terms of adopting a teaching strategy such systems are either tutorial or PMP based. Tutorial-based systems ranges from simple page-turners to MCQ systems with sophisticated Interactive Video technology (see Chapter 3). The development of medical CAL systems requires the collaboration of *knowledgeable doctors* and computer scientists. I stress the need for Knowledgeable Doctors in such collaboration for a number of reasons:

1. The doctor should be *highly experienced in his or her field*. He or she must be able to state his or her problem-solving skills and any other medical information in a structured manner to the non-medical personnel (i.e. the computer specialists) working on the project.

2. The doctor assigned to the project should be one that *accepts the use of computer technology in medicine* (e.g. diagnostic assistant, teaching tool). For many doctors, such

technology is foreign and somewhat threatening. This view is still unfortunately held by some physicians and in particular the older generation.

3. The doctor should *have full knowledge and understanding of the use of computers in medicine and of any known systems in his or her field.* This point is important mainly because on most occasions, the doctor assigned to the project knows nothing about the use of computers in his or her field. Further, any system, no matter how simple and basic, developed by the computer scientists would appear to the doctor as a highly interesting and useful piece of software. This view, although getting less common, is still held by quite a number of doctors in the U.K. In addition, this view can also be seen in the majority of papers published in some medical education journals by medical doctors. One important outcome of this opinion is the difficulty in forming an unbiased view when evaluating such systems.

In this research, we have found and demonstrated that none of the current medical teaching systems are suitable to the General Practice environment.

The critical design issue in building Primary Care systems is the major departure from the well known and established hospital-based systems' concept. A concept that follows a sequential "check-list" approach when dealing with a patient problem. This approach was shown in Figure 3.7. A hospital doctor works by following a pre-defined sequential set of actions that can be summarized as follows:

1. Case information gathering
2. Past histories
3. Physical examination (including laboratory tests)
4. Case diagnosis
5. Case management and therapy

The hospital doctor does not attempt to formulate a differential diagnosis prior to finding the results of all examinations and tests.

A General Practitioner, on the other hand, follows a completely different set of actions that are based on the presenting patient problem. This set of actions combines parallel as well as sequential procedures. The GP follows the model shown in Figure 3.8 for formulating and streamlining his or her differential diagnosis on the basis of each item of relevant information supplied by the patient or observed by the GP.

To summarize, the important medical issue, which is stressed by this research, is the placement of emphasis on patient management rather than disease diagnosis. Although most of the CAL systems developed are hospital based, the few that are aimed at General Practice have also been designed on the hospital method of work. This has the effect of offering little in terms of General Practice education.

A second important issue that emerged from the design of the *P.C.Tutor* is the need for a multi-environment simulation in systems aimed at the Primary Care environment. K.Hodgkin [1985] states that:

> "Primary care is the care which any community considers should be handled by the doctor of first contact."

He points out that the essential point is that the decision on the nature and extent of primary care is made by the community rather than the physician. Therefore, the nature of care differs from one area to another. For example, in rural communities where full Secondary Care is not available, the Primary Care physician has a very wide range of responsibilities. While in rich urban communities, primary care is shared between the General Practitioner and the specialists who frequently wishes to provide primary care. Thus, the GP's role in such community is a confused one as a result of the facilities offered by the specialists. Therefore, this dual care is often neither continuous nor comprehensive.

The varying range of responsibilities forced upon the GP by the cultural, social and economic constraints of the community influences the design of any system that strives to simulate this sophisticated environment.

7.3.2 The Computer Science Viewpoint

The important issue of concern from the computer science point of view is the possible application of the *P.C.Tutor* to domains other than medicine.

A fundamental design and implementation issue of most medical tutoring systems and other ITSs is their ability to be used in domains other than those originally intended. Regarding medical CAL systems, this is due in part to the widespread applicability of the concept of diagnosis and therapy as a problem solving method. One example showing the domain-generality of a medical tutoring system is the use of GUIDON with SACON, an expert system for structural analysis [Clancey, 1987a].

Many problems that can be viewed as the discovery and correction of "errors" can be viewed as diagnosis and therapy, whether the domain is medicine (human, veterinary) or plants-related problems, automobile maintenance, hardware maintenance (electronic or otherwise) and any other domain based on the model shown in Figure 1.2. The model shows the general view of the tutoring strategy adopted in *P.C.Tutor.*

To outline how our system can be used in another domain, consider the following simplified example for computer hardware maintenance and its correspondence to the medical domain. Note that the example does not cover all the hardware maintenance details.

Problems Groups (These correspond to the 19 disease categories -Chapter 4 section 4.2.1.2)

1. Startup Problems
2. Run Problems
3. Display Problems
4. Keyboard Problems
5. Printer/Plotter Problems
6. Floppy Disk Drives Problems
7. Hard Disk Problems

The problems associated with Floppy Disk Drives will be used in this example.

Floppy Disk Drives Problems (These correspond to the diseases or conditions in *P.C.Tutor*)

a. No power supply
b. Defective floppy disk
c. Write protect hole on floppy covered
d. Disk drive out of alignment
e. Broken drive belt
f. Dirty read/write head
g. Drive door not closed
h. Incorrect insertion of diskette
i. Virus infected diskette
.....

Symptoms Areas

a. Drive light
b. Drive sound
c. Floppy disk
d. Drive door
e. Error messages
.....

Figure 7.1 shows the relationship of disk drive problems and symptoms. Therefore, given any number of presenting symptoms, a student should be able to formulate an initial hypothesis as in Figure 7.1.

	Hypothesis								
Symptoms	Drive has no power	Disk Defective	Write Protect Hole Covered	Drive Out of Alignment	Broken Belt	Dirty Head	Disk Inserted Incorrectly	Disk Door not Closed	Virus Infected Disk
Does not boot	****	****		****	****	****	****	****	****
Drive light does not come on	****								
No sound from drive	****								
Drive light comes on		****	****	****	****	****	****	****	****
Drive is noisy		****			****				
Read problems		****		****	****	****	****	****	****
Write problems		****	****	****	****	****	****	****	****
Difficult to insert diskette		****							
Track/sector error message		****		****		****			****
Disk does not work in any drive		****							****
Write protect message			****						
FORMAT does not work			****		****	****	****	****	
Not ready reading message					****		****	****	
Disk works in another drive				****		****			

Figure 7.1 - Hardware Faults by Symptoms

Tests (These correspond to laboratory tests)

a. Perform disk drive diagnostic procedures
b. Perform screen diagnostic procedures
c. Run virus detection software
.....

Management Actions

a. Send machine/drive for repairs
b. Re-insert floppy disk in drive
c. Destroy the faulty diskette
d. Remove write protect cover
e. Switch on the power supply
f. Clean read/write head
g. Close drive door
h. Follow virus treatment procedures
.....

The above simplified example clearly shows the appropriateness of applying our system to other domains. Although no attempt was made to apply the tutoring system to other domains (this was mainly due to lack of time), the theoretical basis on which the domain knowledge used in the system have been defined, should not raise any problems when applying it to other domains. In fact, the transition to other domains should be a trivial process. The system is intended to be used as part of a final year BSc Computer Science project by one of the students who is aiming to modify it for the instruction of automobile maintenance.

7.4 ACHIEVEMENT OF OBJECTIVES

The objectives that were set out by the project team and I, and discussed in Chapter 1 were achieved with varying degrees of success.

In terms of the CTISS project, the aim of building a tutoring system for the General Practice environment has been successfully achieved. The system was highly recommended for use in General Practice departments by a number of General Practice Professors and lecturers.

The success of the tutoring system can be attributed to the framework developed for the design of Primary Care CAL systems. The major shift in the problem-solving strategy adopted by the *P.C.Tutor* has been the main factor in differentiating this system from others in the field of computer-based medical education (see Chapter 3).

However, although my objective of offering the student a self-contained interactive learning environment was not fully achieved due to lack of funds, a reasonable environment is currently offered to the student by the *P.C.Tutor*. In addition, the proposals that I have outlined in section 7.7 for future development represent the main areas for future research and for developing a full learning environment.

7.5 EDUCATIONAL VALUE OF MEDICAL CAL SYSTEMS

The process of evaluating a medical CAL system is a difficult task due to the lack of any formal evaluation procedures.

The most common method adopted by systems developers is the use of two groups of students where one is given instruction through lectures or tutorials, while the other is given instruction by the CAL system (e.g. Murray, 1977; Wigertz *et al.*, 1983; Levine *et al.*, 1987). The subject material supplied to both groups is the same material.

As indicated earlier in Chapter 6, the first evaluation procedure of the *P.C.Tutor* system was made in an informal way. The process concentrated on the human-machine interface and the problem-solving strategy adopted by General Practitioners.

Formal evaluation of the system will be made during the first term of the 1989-1990 academic year. The evaluation process and its results will be the subject of an M.D. thesis to be written by the General Practitioner working on the *P.C.Tutor* project. The method that will be adopted in the evaluation procedure is similar to the one mentioned above.

As with most CAL evaluation procedures, the aim of the *P.C.Tutor* evaluation procedure is to show that at the end, either both groups of students have the same level of understanding of the presented material or the CAL-instructed group is the better one.

The educational value of CAL is far greater than what is available in the classroom. Advocates of CAL argue that "...it is an educational technology which exhibits greater flexibility than any of its predecessors and holds great promise for significantly enhancing the educational experience by providing a rich instructional setting at a low cost." [Association of American Medical Colleges, 1986].

7.6 CURRENT STATE OF THE TUTORING SYSTEM

The first working prototype of the CAL system and part of the intelligent tutor was completed and demonstrated at the 'Computers in Medical Education Conference' (Cardiff, December, 1988). This prototype had the ability to generate a differential diagnosis, provide case data, and respond to any correct or incorrect student actions. The second and last prototype was demonstrated at the CAL89 symposium (Guildford, April, 1989). The prototype had the ability to respond to two levels of students, beginners and advanced. The student was also given the freedom to access the medical data and perform cross referencing operations.

The new expanded version of *P.C.Tutor*, to be named SIMPLE LINCTUS - SIMulated Primary care Learning Environment LINked to a Computerized TUtoring System, which will be developed for use at the Department of General Practice (University of Sheffield) as part of the new curriculum starting in April 1990 will have the following features:

1. An enhanced more realistic simulation of any General Practice environment.

2. A revised student modelling system that provides full feedback to different students' levels of understanding.

3. A more user-friendly interface including colour and graphics.

4. A consultation or diagnostic module that can be accessed by the student.

5. Faster execution through the use of overlays.

7.7 AREAS FOR FUTURE DEVELOPMENT

As with most projects, the CTISS project's two years allocated time has now reached its final stages with the tutoring system far from completed. However, with the one year extension given to this project it is hoped that the system will be completed prior to its introduction in the General Practice course at the Department of General Practice, University of Sheffield.

P.C.Tutor has demonstrated the need for good tutoring systems for use in medical education. The system that we have designed and developed provides the basis for a comprehensive framework of a learning environment for use in future medical teaching/tutoring systems.

The following options or proposals were considered for the future expansion of the complete tutoring system but had to be abandoned due to the unavailability of funds. Nevertheless, such proposals should be taken into account when developing future systems:

1. An electronic text-book on an optical storage media (e.g. CD-ROM, WORM {Write Once Read Many} or MO {Magneto-Optical, very high volume storage media similar to current magnetic media in terms of read and write processes}). Such systems are gaining popularity in the scientific community and some commercial areas. In the case of *P.C.Tutor*, it is possible to implement this feature because of the structure used in the development of the medical database. An electronic text-book offers the student rapid

access to those parts of text which contain just the information he or she requires. Furthermore, this option is less capital intensive than Interactive Video systems. A system of this kind would offer the student the following facilities:

Diagnosis: A differential diagnosis can be found in any text book for most common or important symptoms. However, using an electronic search, the system can generate a differential diagnosis on the basis of any set of findings. The results of such search is far more comprehensive than those obtained from a manual search. In particular, an electronic search usually includes all the diseases or conditions even the rare ones which may be omitted from the textbook.

Clinical Details: Frequently, individual cases raise specific questions about a condition. Finding these details through a manual search can be very difficult. Using an electronic search such action is made easier.

Treatment: Once a diagnosis of a condition is made, it is usually straightforward to find a description of the treatment using any printed text-book. However, for many conditions there are differences of opinion between practitioners which are often not reflected in the main articles. Using an electronic search, references to the management of a condition outside the main article can be located, and differences of opinion are revealed.

Side Effects: Searches for side effects or drug interactions can be conducted for any drug or class of drugs in a quick and efficient way using an electronic textbook.

2. As an addition to the tutoring system and, perhaps to the electronic text-book, the optical storage media can also be used to store ECGs, CT-Scans, X-Rays, photographs of swellings, injuries, skin rashes, etc. for the students to evaluate and learn. As pointed out in Chapter 3, this method is very important in medical education. This type of system, usually known as Interactive Video, is vitally important to medical education.

Furthermore, having an electronic text-book with other visual media also provides a comprehensive interactive learning environment for the doctors of tomorrow.

3. In addition to the suggestions in the previous point, still pictures, etc. can be used in an intelligent way whereby the system applies the Visual Media and Data Interpretation methods discussed in Chapter 2. As an example, given an X-Ray picture, the student has to interpret this X-Ray by pointing out any abnormalities and also justifying his or her actions. In this type of system, all pictures can be built in a hierarchy where any relevant area on that picture once selected can produce a further set of pictures. In addition to the pictures, questioning procedures with full explanation capabilities can be activated by the relevant areas. Such an intelligent system would be of great value in most medical fields (e.g. radiology, pathology, etc.).

4. An Expert (or more precisely an Assistant) system for use by General Practitioners at their surgeries. The need for an assistant can be justified by discussing the many reasons why a doctor might experience difficulty in formulating an appropriate differential diagnosis. It may be that the case involves a rare disease or unusual presentation. Often, such difficulties arise in clinical problems where two or more disease processes are at work, generating a plethora of abnormal findings that can be interpreted in a great variety of ways. It might be, however, argued that patients rarely have two or more significant diseases to be diagnosed. But, even in the case of an uncomplicated instance of disease, there are invariably some discrepant findings that arise due to misrepresentation by the patient, the residual effects of a prior disease episode, or the occurrence of false positive laboratory results. Not uncommonly, apparent discrepancies can be attributed to an incomplete knowledge of the disease on the part of the doctor and/or the biomedical community. Thus the process of sorting patient data into even two partitions (the relevant and the "red herrings") is often difficult for the General Practitioner. Where there is the potential for multiple concurrent diseases, this partitioning becomes ever more difficult.

It is not surprising that even skilled doctors sometimes need assistance in drawing out a comprehensive set of alternative hypotheses of a diagnostic problem.

A medical database of disease entities provides an excellent basis for such a system. Natural Language Processing can be used in this case.

5. Using the tutoring system in the simulation and study of epidemics. In particular, the simulation of disease epidemics helps the student in getting a better view on their effects on a simulated community and on the people according to age and sex. Coping with epidemics is a difficult area for untrained practitioners. Therefore, a simulation of this kind provides the student with an insight into the management and control of such problems.

   Epidemics simulation has not been considered in any of the medical teaching systems.

6. An extensive 'Help' system. This particular system would offer the student the following options:

   a. *Data interpretation* - The purpose of this option is to explain the relevance of known case data.

   b. *Expert Guidance* - This is aimed at the confused student who have no knowledge on what to do next. Such an option should have the ability to interactively guide the student to a point where he or she can continue on his or her own way of problem-solving. An example of a similar option is activated by the CONSULT instruction in THYROID [Steele *et al.*, 1978]

Few of the above proposals are new. While others have been used in some of the hospital-based CAL systems developed in the U.K. The majority of these systems are either tutorial based or PMP type systems. However, a learning environment with the above features and many more (e.g. a word processor for writing reports, papers, etc.), for use in General

Practice teaching should be of great value to undergraduate and postgraduate medical students. With the increasing availability of low cost hardware and software, it is time for such an environment to be developed and used as part of the General Practice curriculums in medical schools. It is also time for educationalists to realize the importance of and the need for introducing computer technology into the medical teaching environment. This would not replace existing teaching practices but would complement them.

REFERENCES

Aikins, J. (1983). Prototypical Knowledge for Expert Systems. *Artificial Intelligence*. 20, 163-210.

Anderson, J.R., and Skwarecki, E. (1986). The Automated Tutoring of Introductory Computer Programming. *Communications of the ACM.* Vol. 29, No. 9, 842-849.

Association of American Medical Colleges (1986). Medical Education in the Information Age. Proceedings of the Symposium on Medical Informatics. *Journal of Medical Education*. June 1986, Vol.61, 113pp.

Barnett, G.O. (1974). The Use of a Computer-Based System to Teach Clinical Problem Solving. In R.W.Stacy and B.D.Waxman (Editors) *Computers in Biomedical Research Volume IV.* Academic Press.

Barnhouse, S. (1988). Knowledge Based Tours: Introducing a Knowledge Based System to a Novice User. *Proceedings of the Intelligent Tutoring Systems (ITS-88) Conference*. Montreal, Canada.

Brown, J.S., Burton, R.R., and de Kleer, J. (1982). Pedagogical, Natural Language and Knowledge Engineering Techniques in SOPHIE I,II,III. In D.H.Sleeman and J.S.Brown (Editors) *Intelligent Tutoring Systems*. Academic Press.

Burton, R.R., and Brown, J.S. (1982). An Investigation of Computer Coaching for Informal Learning Activities. In D.Sleeman and J.S.Brown (Editors) *Intelligent Tutoring Systems*. Academic Press.

Carbonell, J.R. (1970). AI in CAI: An Artificial Intelligence Approach to Computer Assisted Instruction. *IEEE Transactions on Man-Machine Systems,* 11, 190-202.

Carr, B., and Goldstein, I.P. (1977). *Overlays: A theory of modelling for computer aided instruction.* Artificial Intelligence Laboratory Memo 406 (Logo Memo 40). Massachusetts Institute of Technology.

Chambers, J.A., and Sprecher, J.W. (1983). *Computer-Assisted Instruction: Its Use in the Classroom.* Prentice-Hall.

Clancey, W.J., and Letsinger, R. (1984). NEOMYCIN: Reconfiguring a Rule-Based Expert System for Application to Teaching. In W.J.Clancey and E.H.Shortliffe (Editors) *Readings in Medical Artificial Intelligence: The First Decade.* Addison-Wesley.

Clancey, W.J. (1982). Tutoring Rules for Guiding a Case Method Dialogue. In D.H.Sleeman and J.S.Brown (Editors), *Intelligent Tutoring Systems.* Academic Press.

√ **Clancey, W.J. (1987a).** *Knowledge-based tutoring: The GUIDON program.* Cambridge, MA: MIT Press.

Clancey, W.J. (1987b). The Knowledge Engineer as Student: Metacognitive Bases for Asking Good Questions. In H.Mandl and A.M.Lesgold (Editors) *Learning Issues for Intelligent Tutoring Systems.* Springer-Verlag, New York.

√ **Clancey, W.J. (1988).** The Role of Qualitative Models in Instruction. In J.Self (Editor) *Artificial Intelligence and Human Learning: Intelligent Computer-Aided Instruction.* Chapman and Hall.

Clancey, W.J., and Joerger, K. (1988). A Practical Authoring Shell for Apprenticeship Learning. *Proceedings of the Intelligent Tutoring Systems (ITS-88) Conference.* Montreal, Canada.

Clayden, G.S., and Wilson, B. (1988). Computer-Assisted Learning in Medical Education. *Medical Education*, 22, 456-467.

Collins, A., and Brown, J.S. (1987). Cognitive Apprenticeship: Teaching Students the Craft of Reading, Writing, and Mathematics. In L.B.Resnick (Editor) *Cognition and Instruction*. Lawrence Erlbaum Associates.

Computers in Medical Education (Proceedings) (1988). Department of Physiology, University College Cardiff, Cardiff, U.K. December 12th, 1988.

de Dombal, F.T., Hartley, J.R., and Sleeman, D.H. (1969). A Computer Assisted System for Learning Clinical Diagnosis. *Lancet*, 1, 145-148.

de Dombal, F.T. (1971). Teaching Clinical Diagnosis. In L.G.Whitby and W.Lutz (Editors), *Principles & Practice of Medical Computing*. Churchill-Livingstone.

de Dombal, F.T., Adams, I.D., Chan, M., Clifford, P.C., Cooke, W.M., Dallos, V., Edwards, M.H., Hancock, D.M., Hewett, D.J., McIntyre, N., Somerville, P.G., Spiegelhater, D.J., Wellwood, J., and Wilson, D.H. (1986). Computer Aided Diagnosis of Acute Abdominal Pain: A Multicentre Study. *British Medical Journal*, 293, 800-804.

Elsom-Cook, M., Spensley, F., Byerley, P., Brooks, P., Mhende, M., Federici, M., and Scaroni, C. (1988). Using Multiple Teaching Strategies in and ITS. *Proceedings of the Intelligent Tutoring Systems (ITS-88) Conference*. Montreal, Canada.

Elsom-Cook, M. (1988). Guided Discovery Tutoring and Bounded User Modelling. In J.Self (Editor) *Artificial Intelligence and Human Learning: Intelligent Computer-Aided Instruction*. Chapman and Hall.

Florey, C. du V. (1988). Computer-Assisted Learning in British Medical Schools. *Medical Education*, 22,180-182.

Fos, P.J. (1988). An Alternative Approach to Prevention: Computer-Assisted Patient Education. *Journal of Dentistry for Children*. Jan-Feb, 55(1), 43-46.

Fox, J., Glowinski, A., and O'Neil, M. (1987). The Oxford System of Medicine: A prototype information system for primary care. In J.Fox, M.Fieschi, and R.Engelbrecht (Editors) *Proceedings of European Conference on Artificial Intelligence in Medicine (AIME87).* Marseilles, France. Springer-Verlag.

Goldstein, I.P. (1982). The Genetic Graph: A Representation for the Evolution of Procedural Knowledge. In D.Sleeman and J.S.Brown (Editors) *Intelligent Tutoring Systems.* Academic Press.

Grahne, G. (1981). Adaptive Features of a CAL System Based on Information Retrieval. In P.R.Smith (Editor) *Computer Assisted Learning: Selected Papers from the CAL 81 Symposium.* Pergamon Press.

Guardian Newspaper (1989). Baby Talk. In *Computer Guardian: Microfile,* J.Schofield (Editor), July 6th.

Hagglund, S., Elfstrom, J., Holmgren, H., Rosin, O., and Wigertz, O. (1981). Specifying Control and Data in the Design of Educational Software. In P.R.Smith (Editor) *Computer Assisted Learning: Selected Papers from the CAL 81 Symposium.* Pergamon Press.

Harden, R.M. (1983). Preparation and Presentation of Patient-Management Problems (PMPs). *Medical Education,* 17, 256-276.

Harless, W.G., Drennon, G.G., Marxer, J.J., Root, J.A., and Miller, G.E. (1971). CASE: A Computer-Aided Simulation of the Clinical Encounter. *Journal of Medical Education,* 46, 443-448.

Hekmatpour, S., and Ince, D.C. (1986). *Rapid Software Prototyping.* Mathematics Faculty, Open University.

Hodgkin, K. (1985). *Towards Earlier Diagnosis: A Guide to Primary Care.* Churchill Livingstone.

Ingram, D., Jolly, B.C., Hague, L., Grebinnick, K., Hinds, C.J., Conry, B., and Brooks, A. (1989). Use of Interactive Video in Teaching Aspects of Intensive Care, Diagnostic Radiology, and Accident and Emergency Medicine. *Computer Assisted Learning (CAL89) Symposium Handbook and Abstracts Supplement*. Pergamon Press.

Ingram, D. (1988). Medical Educational Comupting. *The CTISS File Number* 7, Oct, pp 7-16.

Johnson, W.L., and Soloway, E. (1987). PROUST: An Automatic Debugger for Pascal Programs. In G.P.Kearsley (Editor) *Artificial Intelligence & Instruction: Applications and Methods*. Addison-Wesley.

Kemmis, S., Atkin, R., and Wright, E. (1977). *How Do Students Learn?.* Working Papers on Computer Assisted Learning. Occational Paper No. 5. Centre for Applied Research in Education, University of East Anglia.

Kimball, R. (1982). A Self-Improving Tutor for Symbolic Integration. In D.Sleeman and J.S.Brown (Editors) *Intelligent Tutoring Systems*. Academic Press.

Levine, R.S., Harold Jones, J., and Morgan, C. (1987). Comparison of Computer-Assisted Learning With Tutorial Teaching in a Group of First-Year Dental Students. *Medical Education*. 21, 305-309.

Mansour, A.A.H. (1989). *Primary Care Tutor: A Medical Tutoring System for Students of General Practice.* University of Sheffield, Department of Computer Science Research Report CS-89-28.

Marmion, B.P., and Lutz, W. (1971). Multiple Choice Questions and the Computer. In L.G.Whitby and W.Lutz (Editors) *Principles & Practice of Medical Computing*. Churchill Livingstone.

McWilliams, A.T. (1988). Introducing Expert Systems to Medical Students Using ESTA, Expert System Teaching Aid. *Medical Education*, 22, 99-103.

Miller, R.A., Pople, Jr., H.E., and Myers, J.D. (1984). INTERNIST-1, An Experimental Computer-Based Diagnostic Consultant for General Internal Medicine. In W.J.Clancey and E.H.Shortliffe (Editors), *Readings in Medical Artificial Intelligence: The First Decade*. Addison Wesley.

Murray, T.S., Cupples, R.W., Barber, J.H., and Hannay, D.R. (1976). Computer-Assisted Learning in Undergraduate Medical Teaching. *Lancet*, 1, 474-476.

Murray, T.S. (1977). *Developing Methods of Evaluation Appropriate to Undergraduate Teaching in General Practice.* Ph.D. Thesis, University of Glasgow.

O'Shea, T., and Self, J.A. (1985). *Learning and Teaching With Computers: Artificial Intelligence in Education*. Harvester.

O'Shea, T. (1982). A Self-Improving Quadratic Tutor. In D.Sleeman and J.S.Brown (Editors) *Intelligent Tutoring Systems*. Academic Press.

Pickell, G.C., Medal, D., Mann, W.S., and Staebler, R.J. (1986). Computerizing Clinical Patient Problems: An Evolving Tool for Medical Education. *Medical Education*, 20, 201-203.

Richer, M.H., and Clancey, W.J. (1988). GUIDON-WATCH: A Graphic Interface for Viewing a Knowledge-Based System. In R.W.Lawler and M.Yazdani (Editors), *Artificial Intelligence and Education Volume One: Learning Environments & Tutoring Systems*. Ablex Publishing.

Riel, M.M., Levin, J.A., and Miller-Souviney, B. (1988). Learning with Interactive Media: Dynamic Support for Students and Teachers. In R.W.Lawler and M.Yazdani (Editors) *Artificial Intelligence and Education Volume One: Learning Environments & Tutoring Systems.* Ablex Publishing.

Self, J.A. (1985). A Perspective on Intelligent Computer-Assisted Learning. *Journal of Computer Assisted Learning.* 1, 159-166.

Skinner, B.F. (1968). *The Technology of Teaching.* New York: Appleton-Century-Crofts.

Shortliffe, E.H. (1976). *Computer-Based Medical Consultations: MYCIN.* American Elsevier, NY.

Shortliffe, E.H., Carlisle Scott, A., Bischoff, B., Bruce Campbell, A., Van Melle, W., and Jacobs, C.D. (1984). ONCOCIN: An Expert System for Oncology Protocol Management. In B.G.Buchanan and E.H.Shortliffe, *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project.* Reading, MA, Addison-Wesley.

Steele, A.A., Davis, P.J., Hoffer, E.P., and Famigletti, K.T. (1978). A Computer-Assisted Instruction (CAI) Program in Diseases of the Thyroid Gland (THYROID). *Computers and Biomedical Research.* 11, 133-146.

Taylor, T.R. (1975). Computer Assisted Learning in Clinical Decision-Making. In R.Hooper and I.Toye (Editors) *Computer Assisted Learning in the United Kingdom: Some Case Studies.*

Verbeek, H.A. (1987). Self-Instruction Through Patient Simulation By Computer. *Medical Education*, 1987, 21, 10-14.

Wigertz, O., Elfstrom, J., Hagglund, S., and Rosin, O. (1983). Computer-Assisted Training in Patient Management and Clinical Decision Making: Methodological Approach and Experiences. In J.C.Pages, A.H.Levy, F.Gremy and J.Anderson (Editors) *Meeting the Challenge: Informatics & Medical Education.* Elsevier Science Publishers B.V. (North-Holland).

Woolf, B. (1984). *Context-dependent planning in a machine tutor.* Ph.D. Dissertation. Computer and Information Sciences, University of Massachusetts, Amherst, MA.

Woolf, B., and McDonald, D.D. (1984). Building a Computer Tutor: Design Issues. *IEEE Computer.* September 1984, 61-73.

Woolf, B. (1988). Representing Complex Knowledge in an Intelligent Machine Tutor. In J.Self (Editor), *Artificial Intelligence and Human Learning: Intelligent Computer-Aided Instruction.* Chapman and Hall.

Woolf, B. (1988). 20 Years in the Trenches: What Have We Learned?. *Proceedings of the Intelligent Tutoring Systems (ITS-88) Conference*. Montreal, Canada.

APPENDIX A

THE STRUCTURING OF DATA IN THE MEDICAL DATABASE

In this appendix, I will be discussing in brief the methods used to structure the medical data in the database.

Parts of the medical data concerning each disease or medical condition parameters, which have been specified by the General Practitioner working on the CTISS project, are constructed using a special taxonomy based on hierarchical structures. These include all the symptoms, clinical signs, laboratory tests, and management interventions. The following sections briefly discusses these hierarchical structures.*

A.1 SYMPTOMS

All symptoms have been categorized into four main areas. These are:

1. Altered Sensorium
2. Physical Dysfunction
3. Self Perceived Signs
4. Physical and Life Events

Each area is divided in turn into sub-areas or sub-levels representing specific symptoms (e.g pain, itching, etc.), and some general symptoms (e.g. speech problems, memory problems, etc.). The latter consists of another sub-level of specific symptoms (e.g. remembers only a few events, speech is unintelligible, etc.).

---

* The structures discussed in this appendix apply to the current prototype of the medical system. Some changes are expected to be made to the final version of the system.

Body areas (or more precisely body sites) are defined into groups according to their overall area. For example:

Group A: Head, Neck, Face, Ear, Lips, etc.
Group B: Shoulder, Arm, Hand, etc.
. . .
. . .

To construct a normal English sentence as spoken by a non-medical person, several qualifiers are required in addition to the specific symptoms and body areas. These include:

1. *Site Qualifiers* - e.g. Right, Left, Middle, Lower, Upper, etc.

2. *Character Qualifiers* - e.g. Sharp, Dull, Burning, Hard, Hot, Severe, Red, etc.

3. *Temporal Qualifiers* - e.g. Seconds, Minutes, Days, today, Occasionally, etc.

The above qualifiers are connected using connectors such as *of the*, *in the*, *on the*, *for the last*, *every*, etc. when sentences are constructed.

The main advantages in using this particular sentence construction system are:

1. Interpreting the sentences is made easier and quicker than interpreting those entered in a free text system - text interpretation is a matter of looking through a symbols table rather than full natural language processing.

2. Addition of new parts is a matter of adding them to the appropriate level in the hierarchy.

The other type of symptoms used in the system are entered as complete sentences for the expert author to use when defining a particular disease or medical condition (e.g. feels unhappy most of the time, has a handicapped child, has been divorced, etc.).

A.2 CLINICAL SIGNS AND EXAMINATIONS

A similar hierarchical structure to that used in the design of the symptoms is employed with the clinical signs and examinations. The root options in this structure are:

1. General Condition (e.g. restless, short of breath, etc.)
2. Examination of Body Area
3. Auscultation (e.g. breath sound, heart sound, etc.)
4. Percussion

Each of the above options has an underlying unique code. The option Examination of Body Area has a sub-level consisting of all body systems (e.g. Upper Respiratory Tract - URT, Cardiovascular system, etc). Each body system consists of another detailed level. For example, the URT consists of the mouth, teeth & gums, oropharynx, middle ear, etc.

All body systems with the exception to the Cardiovascular system can be Inspected or Palpated. The possible results of the Inspection or Palpation of the body systems are stored as simple template sentences for the expert GP to select and append the appropriate body system details (e.g. a swelling of the ..., tenderness in the ..., etc.).

Finally, the Cardiovascular system consists of three sub-levels: Pulse, Apex Beat, and Blood Pressure. Each sub-level consists of further level(s) containing all the possible results associated with the main sub-levels (e.g. Pulse - Regular, Rapid, etc.).

A.3 LABORATORY TESTS

As mentioned earlier in this thesis, the laboratory tests are classified into two categories: (1) tests which are performed by the GP at his or her surgery, and (2) tests which are made at outside laboratories.

The first category consists of three specific urinalysis tests.

Each test in the second category has a specific code associated with it. The tests are grouped into five overall categories. These are:

1. Examination of Body Fluids
2. Plain X-Ray
3. Contrast X-Ray
4. Ultrasound
5. Examination of Tissue

Each category consists of one, two or three sub-levels. For example, Examination of Body Fluids has a further level with three specific types of tests. These are:

* Blood for Haematology
* Blood for Biochemistry
* Blood for Enzymes
* Urinalysis
* Specimens for Culture

Considering Blood for Biochemistry, a further level is also associated with it. This consists of:

** Urea and Electrolytes
** Liver Function Tests
** Uric Acid

Although these tests are requested by the GP in this format, the actual results of some of them are supplied by the laboratories as finite vales, e.g. Urea and Electrolytes are given as:

Bicarbonate,	Chloride,	Potassium,
Sodium,	Calcium,	Iron,
Creatinine.		

To summarize, the structure used for the laboratory tests is a simple hierarchy of menus.

All laboratory tests are defined according to the current common practice that exists between GPs and outside laboratories.

A.4 MANAGEMENT INTERVENTIONS

The structure that is adopted for storing and accessing the management interventions is very similar to that used with the laboratory tests. Therefore, I will not attempt to discuss it any further.

A.5 CONCLUDING REMARKS

The hierarchical structures adopted in the design of the medical database have proved to be very effective in terms of easy access to the medical data, easy maintenance of the database, and the simple expansion of the database. In addition, the size of the system in terms of bytes has been reduced due to the use of symbol tables.

APPENDIX B

MEDICAL DATABASE CONTENT

This appendix shows the madical data used in the testing of the *P.C.Tutor* system. This data have been entered by the General Practitioner assigned to the CTISS project. The Prevalence figures were obtained from Hodgkin [1985]. Results of Laboratory Tests were obtained from various published medical tables.

The information associated with each of the diseases and problems in this appendix are not finalised.

VIRAL PHARYNGITIS
(Sore Throat)

Disease Group: Infective And Parasitic Diseases

Symptoms:
*Mild soreness in the throat for the last 36 hours
Shivering

Clinical Signs/Examinations:
Redness of the oropharynx
Tender swelling of the cervical lymph nodes

Laboratory Tests:
Blood For Haematology - E.S.R.: 6,15
Blood For Haematology - Monospot: positive

Management Plan:

Optional:
Simple Analgesic
(Paracetamol is the drug of choice for symptomatic relief.)

Throat Swab
(A throat swab can help differentiate the diagnosis, but the illness is usually over before the result returns.)

Unnecessary:
Penicillin
(Treatment of simple sore throat with an antibiotic has no benefit to the patient but could be argued to be an acceptable treatment where the diagnosis is in doubt and the symptoms severe.)

Graph Type	Self Limiting Disease
Time 1	1 - 3 days
Time 2	2 - 5 days
Sex Ratios (M:F)	1:1
Age Range	2 - 85
Severity	Minor
Prevalence	91.3 in 1000 patients

INFLUENZA
(Flu)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Shivering today
*Dull aching in the lower back for the last 2 days
Coughs and brings up clear spit for the last 4 days
Frequent moderate soreness in the chest
Sweating for the last 2 days
Sharp soreness in the throat today
Moderate aching in the head for the last 3 days

Clinical Signs/Examinations:

Moderately distressed
Moderately pyrexial
Redness of the oropharynx

Laboratory Tests:

Blood For Haematology - E.S.R.: 15,30
Blood For Haematology - White Cell Count: 7000,15000
Blood For Haematology - Neutrophils: 3000,10000
Blood For Haematology - Lymphocytes: 3000,6000

Management Plan:

<u>Optional:</u>

Simple Analgesic
(Relief of the discomfort of pyrexia and the somatic aches and pains are of paramount importance. Reducing the duration of the disease is not possible so the relief of symptoms is the only option.)

<u>Unnecessary:</u>

Penicillin Esters
(Treatment with antibiotics is generally unhelpful. Patients who are in a high risk category (the elderly or astmatics for instance) may benefit from prophyllactic antibiotic treatment.)

Tetracycline

Sulphonamide

Co Trimoxazole

Trimethoprim

Graph Type Self Limiting Disease
Time 1 2 - 4 days
Time 2 4 - 10 days
Sex Ratios (M:F) 1:1
Age Range 25 - 85
Severity Moderate
Prevalence 10 in 1000 patients

OTITIS MEDIA

(Middle Ear Infection, Infection of the Middle Ear, Earache)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Sharp earache today
Sweating
Shivering today

Clinical Signs/Examinations:

Moderately pyrexial
Redness of the tympanic membrane

Laboratory Tests:

(no labtests present)

Management Plan:

Mandatory:

Simple Analgesic

(This very painful condition can cause much discomfort for the patient and disturb small children a great deal. Adequate analgesia is essential.)

Penicillin Esters

(A penicillin ester is probably the treatment of choice in this condition, with a cephalosporin reserved for those allergic to penicillin.)

Optional:

Cephalosporin

(Considered by some to be a first line treatment, but consensus probably reserves use for patients with penicillin allergy.)

Decongestant

(The use of decongestants is not mandatory, but a preferred additional treatment for many GPs.)

Graph Type	Self Limiting Disease
Time 1	8 - 12 hours
Time 2	24 - 48 hours
Sex Ratios (M:F)	1:1
Age Range	1 - 7
Severity	Minor
Prevalence	66 in 1000 patients

GLANDULAR FEVER

(No synonyms)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Moderate sore throat for the last 3 days
Sweating
Dull aching in the lower back for the last 5 days

Clinical Signs/Examinations:

Moderately distressed
Very pyrexial
Redness of the soft palate
Tender swelling of the cervical lymph nodes
Tender swelling of the axillary lymph nodes
Tender swelling of the submandibular lymph nodes
Painless swelling of the spleen

Laboratory Tests:

Blood For Haematology - E.S.R.: 14,15
Blood For Haematology - White Cell Count: 10000,11000
Blood For Haematology - Monospot: positive
Blood For Haematology - Neutrophils: 5000,8000

Management Plan:

Mandatory:

Simple Analgesic
(All patients should receive symptomatic relief. Paracetamol for relief of fever and pain, with soluble aspirin gargles for a sore throat.)

Optional:

Penicillin
(Acutely ill patients in whom a clinical differentiation between tonsillitis and Glandular Fever cannot be made may well be optimally treated with penicillin.)

Harmful:

Penicillin Esters
(NEVER give ampicillin or derivatives for sore throats as the complication of exfoliative dermatitis seen with these drugs in Glandular Fever can be lethal !!)

Graph Type	Self Limiting Disease
Time 1	1 - 2 weeks
Time 2	3 - 5 weeks
Sex Ratios (M:F)	1:1
Age Range	13 - 25
Severity	Moderate
Prevalence	1.3 in 1000 patients

RUBELLA

(German Measles)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Skin rash of the lower back for the last 18 hours
Sweating at night

Clinical Signs/Examinations:

Slightly pyrexial
Smooth swelling of the occipital lymph nodes
Erythematous maculo-papular rash of the chest
Erythematous maculo-papular rash of the upper back
Erythematous maculo-papular rash of the lower back

Laboratory Tests:

(no labtests present)

Management Plan:

<u>Optional:</u>

Simple Analgesic
(Fever is mild but paracetamol may be helpful.)

Graph Type	Self Limiting Disease
Time 1	1 - 2 days
Time 2	2 - 4 days
Sex Ratios (M:F)	1:1
Age Range	1 - 30
Severity	Minor
Prevalence	5.1 in 1000 patients

MEASLES

(No synonyms)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Has a dry cough for the last 2 days
*Skin rash of the chest today
Soreness in the left eye today
Shivering today

Clinical Signs/Examinations:

Moderately pyrexial
Whiteness of the soft palate
Redness of the oropharynx
Tender swelling of the cervical lymph nodes
Macular rash of the chest
Macular rash of the lower back
Macular rash of the upper back

Laboratory Tests:

(no labtests present)

Management Plan:

Mandatory:

Simple Analgesic
(Measles can make children very miserable and paracetamol can help reduce morbidity.)

Optional:

Penicillin Esters
(Complications of Otitis Media and chest infections may need antibiotics.)

Graph Type	Self Limiting Disease
Time 1	3 - 6 days
Time 2	7 - 14 days
Sex Ratios (M:F)	1:1
Age Range	1 - 15
Severity	Minor
Prevalence	4.5 in 1000 patients

ACUTE TRACHEITIS
(Tracheitis)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Coughs and brings up green spit for the last 3 days
Dull soreness in the middle of the chest for the last 3 days
Sweating for the last 2 days

Clinical Signs/Examinations:

Slightly pyrexial

Laboratory Tests:

(no labtests present)

Management Plan:

(no interventions specified by the author - temporary state)

Graph Type	Self Limiting Disease
Time 1	12 - 24 hours
Time 2	24 - 72 hours
Sex Ratios (M:F)	1:1
Age Range	25 - 70
Severity	Minor
Prevalence	30 in 1000 patients

HAND FOOT AND MOUTH DISEASE

(Coxsackie A Disease, Coxsackie A, Hand, Foot and Mouth Disease)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Skin rash of the left hand today
*Skin rash of the right hand today
*Soreness in the mouth today
Skin rash of the left foot today
Skin rash of the right foot today

Clinical Signs/Examinations:

Vesicular rash of the tongue
Vesicular rash of the hand
Vesicular rash of the foot

Laboratory Tests:

(no labtests present)

Management Plan:

Optional:

Simple Analgesic
(This largely benign condition has slight morbidity, and symptoms can be relieved with paracetamol.)

Graph type	Self Limiting Disease
Time 1	12 - 24 hours
Time 2	24 - 72 hours
Sex Ratios (M:F)	1:1
Age Range	2 - 10
Severity	Minor
Prevalence	5 in 1000 patients

BORNHOLM DISEASE

(Epidemic Intercostal Myalgia, Epidemic Myalgia, Coxsackie B, Coxsackie B infection, Devil's Grip)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Severe pain in the left side of the chest for the last 2 days
Sweating today
Shivering today
Mild aching in the whole body today

Clinical Signs/Examinations:

Slightly distressed
Quite pyrexial
Tenderness of the left ninth rib

Laboratory Tests:

(no labtests present)

Management Plan:

<u>Mandatory:</u>

Compound Analgesic
(This is a very painful condition requiring adequate relief of pain.)

<u>Optional:</u>

Plain X-Ray, Chest
(Differentiating this condition from rib fracture or an underlying pleurisy can be difficult, and is helped by radiology on occasions.)

Graph type	Self Limiting Disease
Time 1	2 - 5 days
Time 2	3 - 9 days
Sex Ratios (M:F)	1:1
Age Range	20 - 50
Severity	Moderate
Prevalence	0.5 in 1000 patients

INFECTIOUS HEPATITIS

(Short Incubation Hepatitis, MS1 Hepatitis, Hepatitis A, Infectious Jaundice)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Always vomits for the last 3 days
*Colouration of the whole body today
Opens bowels with difficulty
Dull pain in the abdomen for the last 2 days

Clinical Signs/Examinations:

Slightly pyrexial
Yellowness of the chest
Yellowness of the abdomen
Yellowness of the upper back
Yellowness of the lower back

Laboratory Tests:

Blood For Biochemistry - Bilirubin: 15,75.5
Blood For Enzymes - SGOT: 150,250
Blood For Enzymes - SGPT (ALT): 35,65
Blood For Enzymes - Alkaline Phosphatase: 300,500
Blood For Enzymes - Creatine Kinase: 50,90

Management Plan:

Mandatory:

Blood for Enzymes
(Liver function may be grossly disturbed and enzyme profiles can also be very useful.)

Blood for Biochemistry
(Sub-clinical jaundice may occur and can be elucidated by performing a serum bilirubin.)

Optional:

Urinalysis
(The presence of urobilinogen can confirm jaundice.)

Graph type Self Limiting Disease
Time 1 3 - 14 days
Time 2 7 - 28 days
Sex Ratios (M:F) 1:1
Age Range 10 - 50
Severity Minor
Prevalence 1 in 1000 patients

CHICKENPOX

(No synonyms)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Itching in the whole body
Itching in the scalp
Skin rash of the whole body for the last 1 days
Skin rash of the scalp

Clinical Signs/Examinations:

Pustular vesicular rash of the chest
Pustular vesicular rash of the face
Slightly pyrexial
Pustular vesicular rash of the upper back

Laboratory Tests:

(no labtests present)

Management Plan:

Optional:

Simple Analgesic
(Paracetamol may help a fever and can reduce pruritis)

Emollient & Barrier Preparations
(Traditionally calamine lotion has been used to sooth the rash, may be of dubious clinical benefit but often soothes the carers more.)

Unnecessary:

Anti Infective Preparations
(You do not treat with antiviral preparations.)

Graph type	Self Limiting Disease
Time 1	2 - 5 days
Time 2	7 - 14 days
Sex Ratios (M:F)	1:1
Age Range	3 - 25
Severity	Minor
Prevalence	4.2 in 1000 patients

HERPES ZOSTER
(Shingles)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Burning pain in the right lower chest for the last 5 days
Skin rash of the right lower chest today

Clinical Signs/Examinations:

Slightly distressed
Slightly pyrexial
Vesicular rash of the right T8 dermatome

Laboratory Tests:

(no labtests present)

Management Plan:

Mandatory:

Anti Infective Preparations

(Topical idoxuridine has been shown to reduce post herpetic neuralgia but oral acyclovir has no advantages and is more expensive. Ophthalmic complications probably ought to be seen by an ophthalmologist.)

Optional:

Compound Analgesic

(Herpes Zoster is a very painful condition, with neuralgia often extending for years after the primary lesion. Analgesia can be helpful but is often inneffective.)

Antidepressant

(Pain relief in this condition is notoriously difficult, but the addition of an antidepressant may improve both the pain control and any underlying depressive symptoms, which is often a feature of this condition.)

Graph type	Self Limiting Disease
Time 1	2 - 6 days
Time 2	7 - 28 days
Sex Ratios (M:F)	1:1
Age Range	25 - 75
Severity	Moderate
Prevalence	4.1 in 1000 patients

HERPES SIMPLEX

(Cold Sores)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Itching in the lips for the last 12 hours
*Skin rash of the lips
Has been ill for the last 3 days

Clinical Signs/Examinations:

Vesicular rash of the lips
Tender swelling of the submandibular lymph nodes

Laboratory Tests:

(no labtests present)

Management Plan:

Unnecessary:

Anti Infective Preparations

(As a rule the application of idoxuridine is unnecessary except in the most severe of cases or in the immunocompromised patient. Ultra violet initiated attacks can be aborted by topical acyclovir.)

Graph type	Self Limiting Disease
Time 1	1 - 3 days
Time 2	4 - 7 days
Sex Ratios (M:F)	1:1
Age Range	1 - 85
Severity	Minor
Prevalence	11.5 in 1000 patients

WHOOPING COUGH

(No synonyms)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Coughs and brings up clear spit at night
*Always has a dry cough
Occasional vomits
Mild swelling of the eye

Clinical Signs/Examinations:

Moderately distressed
Slightly pyrexial
Crepitations at both bases of the lungs

Laboratory Tests:

Specimens For Culture - Pernasal Swab: haemophilus pertussis isolated

Management Plan:

Optional:

Erythromycin
(Treatment is only helpful in the catarrhal stage, in the first 14 days.)

Linctus
(A cough supressant, pholcodine linctus, can help in the paroxysmal stage.)

Respiratory Physician
(Admission to a paediatric unit may be needed for severe infections, particularly of the very young. Seldom required for patients over the age of 5 years.)

Graph type Self Limiting Disease
Time 1 1 - 3 weeks
Time 2 2 - 14 weeks
Sex Ratios (M:F) 1:1
Age Range 0 - 12
Severity Moderate
Prevalence 2.4 in 1000 patients

SCARLET FEVER

(No synonyms)

Disease Group: Infective And Parasitic Diseases

Symptoms:

*Skin rash of the whole body for the last 12 hours
Mild headache
Vomits food
Soreness in the throat for the last 36 hours

Clinical Signs/Examinations:

Moderately pyrexial
Redness of the oropharynx
Whiteness of the oropharynx
Erythematous maculo-papular rash of the chest
Tender swelling of the cervical lymph nodes
Erythematous maculo-papular rash of the upper back
Erythematous maculo-papular rash of the lower back

Laboratory Tests:

Specimens For Culture - Throat Swab: streptococcus pyogenes isolated

Management Plan:

Mandatory:

Penicillin
(Streptococcus is the causal organism and penicillin V remains the drug of choice.)

Optional:

Simple Analgesic
(Paracetamol to reduce fever. Soluble aspirin gargles can help the sore throat.)

Unnecessary:

Blood for Haematology
(Erythrogenic toxin can be found and confirms the diagnosis, if you feel you need it.)

Graph type	Self Limiting Disease
Time 1	2 - 5 days
Time 2	5 - 10 days
Sex Ratios (M:F)	1:1
Age Range	5 - 12
Severity	Minor
Prevalence	1.6 in 1000 patients

APHTHOUS ULCERS

(No synonyms)

Disease Group: Diseases Of The Digestive System

Symptoms:

*Irregular soreness in the mouth which lasts for 48 hours
Occasional mild sore tongue which lasts for 2 days

Clinical Signs/Examinations:

Whiteness of the mouth

Laboratory Tests:

(no labtests present)

Management Plan:

<u>Optional:</u>

Corticosteroids
(Hydrocortisone pellets can often help the pain of these ulcers.)

Graph type	Self Limiting Disease
Time 1	24 - 48 hours
Time 2	48 - 96 hours
Sex Ratios (M:F)	1:1
Age Range	5 - 55
Severity	Minor
Prevalence	10.5 in 1000 patients

OESOPHAGITIS

(No synonyms)

Disease Group: Diseases Of The Digestive System

Symptoms:

*Burning pain in the middle of the chest at night
Diminished appetite for the last 2 weeks

Clinical Signs/Examinations:

Tenderness of the epigastrium

Laboratory Tests:

(no labtests present)

Management Plan:

Optional:

Alginate Raft

Simple Antacid

Graph type	Self Limiting Disease
Time 1	3 - 7 days
Time 2	7 - 14 days
Sex Ratios (M:F)	1:2
Age Range	40 - 65
Severity	Moderate
Prevalence	1.5 in 1000 patients

INGUINAL HERNIA

(Hernia, Rupture)

Disease Group: Diseases Of The Digestive System

Symptoms:

*Occasional soft swelling of the left side of the scrotum
Had a fall outside
Dull aching in the left scrotum for the last 6 weeks

Clinical Signs/Examinations:

Swelling in the scrotum
Smooth swelling in the scrotum
Smooth swelling in the left groin
Swelling in the left groin

Laboratory Tests:

(no labtests present)

Management Plan:

<u>Optional:</u>

General Surgeon
(Surgical repair is probably the best treatment, trusses and belts may be preferred by the elderly or infirm. Esentially the patient's choice.)

Graph type	Disease Remains At Constant State
Time 1	1 - 24 hours
Time 2	(Not Applicable)
Sex Ratios (M:F)	10:1
Age Range	35 - 75
Severity	Moderate
Prevalence	5.5 in 1000 patients

STRANGULATED INGUINAL HERNIA

(No synonyms)

Disease Group: Diseases Of The Digestive System

Symptoms:

*Hard swelling of the right scrotum for the last 12 hours
Colouration of the right scrotum for the last 8 hours
Occasional soft swelling of the right scrotum for the last 6 months

Clinical Signs/Examinations:

Moderately distressed
Swelling in the scrotum
Redness of the scrotum
Painful swelling in the scrotum
Swelling in the right groin
Hot swelling in the right groin
Painful swelling in the right groin

Laboratory Tests:

(no labtests present)

Management Plan:

Mandatory:

General Surgeon

(Strangulation of a hernia seems a real possibility. Reduction of a strangulation of short duration is possible, but beware a non-viable loop and be prepared for a recall if there is a perforation !!)

Graph type	Disease Continues To Worsen With Time
Time 1	1 - 72 hours
Time 2	(Not Applicable)
Sex Ratios (M:F)	10:1
Age Range	35 - 65
Severity	Fatal
Prevalence	5.5 in 1000 patients

DUODENAL ULCER

(Duodenal Ulceration, Petic Duodenal Ulcer, Ulcer of the Duodenum)

Disease Group: Diseases Of The Digestive System

Symptoms:

*Frequent gripping aching in the top of the abdomen
Vomits after some meals for the last 6 weeks
Sleeps lightly for the last 4 weeks
Vomits dark blood sometimes

Clinical Signs/Examinations:

Tenderness in the epigastrium

Laboratory Tests:

Blood For Haematology - Haemoglobin: 8.5,11.0
Contrast X-Rays - Barium Meal: There is an active ulcer crater within a deformed duodenum

Management Plan:

Mandatory:

Contrast X-Ray, Barium Meal
(A barium meal is essential to elucidate the exact nature of the peptic disease, and whether ulceration is present. Appropritae treatment {with H2 antagonist} should only be commenced after Barium studies {or endoscopy}.)

Optional:

Simple Antacid
(A first line treatement until proper diagnosis has been made.)

H2 Receptor Blocker
(To be indicated only after confirmatory radiology or endoscopy.)

Graph type	Disease Remains At Constant State
Time 1	1 - 3 months
Time 2	(Not Applicable)
Sex Ratios (M:F)	4:1
Age Range	20 - 50
Severity	Moderate
Prevalence	4 in 1000 patients

HIATUS HERNIA

(Diaphragmatic Hernia)

Disease Group: Diseases Of The Digestive System

Symptoms:

*Frequent severe soreness in the middle of the chest which lasts for 2 days
Occasional burning pain in the top of the abdomen which lasts for 12 hours
Vomits after some meals
Loss of appetite for the last 4 weeks
Has difficulty swallowing sometimes

Clinical Signs/Examinations:

Slightly obese

Laboratory Tests:

Contrast X-Rays - Barium Meal: shows a sliding hiatus hernia with free oesophageal reflux

Management Plan:

Mandatory:

Alginate Raft
(These moderately severe symptoms must be relieved.)

Contrast X-Ray, Barium Meal
(Recurrent episodes of dyspepsia should always be investigated, but endoscopy may be preferred.)

Optional:

H2 Receptor Blocker
(Reduction of gastric acidity can be helpful, although bile reflux may not be helped by these drugs.)

Dopamine Antagonists (Motility Stims)
(Metoclopamide and domperidone can improve sphincter tone and reduce symptoms as well as improve nausea and vomiting.)

Gastroenterologist
(May be the only way to get endoscopy. Difficult cases which a GP finds difficult to control can be referred or if the diagnosis is in doubt.)

Graph type	Disease Remains At Constant State
Time 1	12 - 48 hours
Time 2	(Not Applicable)
Sex Ratios (M:F)	1:2
Age Range	40 - 65
Severity	Moderate
Prevalence	1.5 in 1000 patients

CARCINOMA OF BRONCHUS

(Bronchial Neoplasia, Bronchial Carcinoma, Bronchial Cancer, Cancer of the Lung, Lung Cancer)

Disease Group: Neoplasms

Symptoms:

*Coughs and brings up blood stained spit for the last 2 weeks
Dull aching in the chest for the last 3 months
Short of breath walking uphill sometimes
Sweating at night for the last 10 weeks
Lump on the left side of the neck

Clinical Signs/Examinations:

Slightly cachectic
Swelling in the hypochondrium
Bronchial breathing at the left mid-zone of the lungs
Cold tenderness of the wrist joint
Craggy swelling of the liver
Fixed swelling of the axillary lymph nodes
Dull to percussion at the left mid-zone of the lungs

Laboratory Tests:

Blood For Haematology - E.S.R.: 20,75
Blood For Haematology - Haemoglobin: 9,11.0
Plain X_Rays - Chest: mass at the left hilum with basal atelectasis
Plain X_Rays - Hand: periosteal elevation of the distal radius

Management Plan:

Mandatory:

Plain X-Ray, Chest
Respiratory Physician
(Haemoptysis with other positive symptoms MUST be investigated.)

Optional:

Plain X-Ray, Hand
(Pulmonary osteodystrophy can cause pain in wrist and ankle.)

Blood for Haematology
(A raised ESR can be helpful. There may be anaemia which could be contributing to symptoms of general malaise and breathlessness.)

Graph type	Disease Continues To Worsen With Time
Time 1	1 - 24 months
Time 2	(Not Applicable)
Sex Ratios (M:F)	2:1
Age Range	45 - 65
Severity	Fatal
Prevalence	1.0 in 1000 patients

CARCINOMA OF STOMACH

(Neoplasia of the Stomach, Stomach Cancer, Cancer of the Stomach)

Disease Group: Neoplasms

Symptoms:

*Dull pain in the abdomen for the last 10 days which lasts for 4 hours
Diminished appetite for the last 6 weeks
Vomits after some meals for the last 4 weeks
Rapid weight loss for the last 6 weeks

Clinical Signs/Examinations:

Moderately cachectic
Swelling in the epigastrium
Craggy mass in the epigastrium

Laboratory Tests:

Blood For Haematology - E.S.R.: 25,45
Blood For Haematology - Haemoglobin: 8,12.0
Contrast X-Rays - Barium Meal: Ulcer on the lesser curve. High probability of carcinoma
Ultra Sound - US Scan, Abdomen for Gall Bladder: Gall bladder normal, but several large defects in liver

Management Plan:

Mandatory:

General Surgeon
(A referral for further investigation is essential, the choice of either surgeon or gastroenterologist depends upon the evidence of operability.)

Optional:

Simple Antacid
(Often relieves the pain, at least for a few weeks.)

Dopamine Antagonists (Motility Stims)
(The nausea can be unremitting in this condition, but the symptoms can be helped by this class of drugs.)

Contrast X-Ray, Barium Meal
(A contrast x-ray (or endoscopy) is essential in this sort of case, but it may be more helpful for this to be done through a hospital specialist.)

Gastroenterologist
(A gastroenterologist can confirm the diagnosis in a case where distant metastasis is confirmed and no surgery is contemplated.)

Harmful:

H2 Receptor Blocker
(These drugs can certainly relieve the pain for a considerable time, but should not be used until a positive diagnosis has been established.)

Graph type Disease Continues To Worsen With Time
Time 1 3 - 6 months
Time 2 (Not Applicable)
Sex Ratios (M:F) 3:1

Age Range	40 - 70
Severity	Fatal
Prevalence	0.9 in 1000 patients

CARCINOMA OF OESOPHAGUS

(Oesophageal Neoplasia, Oesophageal Carcinoma, Oesophageal Cancer, Cancer of the Gullet, Cancer of the Oesophagus)

Disease Group: Neoplasms

Symptoms:

*Dull soreness in the middle of the chest for the last 4 weeks
Can only swallow liquids for the last 5 days
Diminished appetite for the last 10 weeks

Clinical Signs/Examinations:

Moderately cachectic
Tenderness of the epigastrium
Craggy swelling of the liver

Laboratory Tests:

Plain X-Rays - Chest: Slight hilar prominence on the right side
Contrast X-Rays - Barium Meal: Hold up of barium with an ulcerated stricture at T5 level
Blood For Haematology - E.S.R.: 25,60
Blood For Haematology - Haemoglobin: 9,12
Blood For Enzymes - SGOT: 15,30.0
Blood For Enzymes - SGPT (ALT): 10,25.0
Blood For Enzymes - Alkaline Phosphatase: 50,150
Blood For Enzymes - Creatine Kinase: 60,110
Blood For Biochemistry - Bilirubin: 12,20

Management Plan:

<u>Mandatory:</u>

Alginate Raft
(Important to relieve symptoms and alginates are probably the best way.)

Gastroenterologist
(Probably no excuse for delaying a referral in a patient like this.)

<u>Optional:</u>

Simple Antacid

Contrast X-Ray, Barium Meal
(Barium studies and/or endoscopy will have to be performed, and you may wish to begin this process yourself, but referral to a gastroenterologist is probably the most useful course.)

Cardio-Thoracic Surgeon
(Operable lesions are dealt with by thoracic surgeons through a combined thoraco-abdominal incision, but direct referral to a cardio-thoracic unit by a general practitioner is an unusual event.)

General Surgeon
(Some general surgeons will deal with these lesions after direct referral from a general practitioner. The choice of such a referral rests with the GP concerned.)

<u>Harmful:</u>

H2 Receptor Blocker

(H2 blockers will certainly help reduce pain, but be warned that they may mask the symptoms in a potentially lethal disease and delay diagnosis.)

Graph type	Disease Continues To Worsen With Time
Time 1	3 - 24 months
Time 2	(Not Applicable)
Sex Ratios (M:F)	2:1
Age Range	50 - 90
Severity	Fatal
Prevalence	0.1 in 1000 patients

PREGNANCY

(No synonyms)

Disease Group: Family Planning, Complications of Pregnancy, Childbirth and the Puerperium

Symptoms:

*Has no periods for the last 3 months
Vomits in the morning for the last 4 weeks
Has to pass urine every 30 minutes

Clinical Signs/Examinations:

Smooth swelling in the hypogastrium

Laboratory Tests:

Urinalysis - Pregnancy Test: positive
Ultra Sound - US Scan, Foetal Heart Only: Foetal heart audible

Management Plan:

Mandatory:

Obstetrician

(Although shared antenatal care is the rule, a hospital booking on confirmation of pregnancy is mandatory. Home confinements do not require a specialist referral, although home delivery is not a popular option at present.)

Patient To Sign FP24 (Maternity Services Claim)

(There is no medical reason for this, but payment under the Statement of Fees and Allowances will only by made with a patient's signature. Better to get into a routine of obtaining a signature at confirmation of pregnancy, which also avoids problems of claims after miscarriage.)

Optional:

Pregnancy test

(A pregnancy test may be requested by the patient, but a vaginal examination can confirm pregnancy at 8-12 weeks gestation.)

Ultrasound, Foetal Heart Only

(A simple ultrasound assessment of foetal heart, if available, can easily demonstrate a viable pregnancy from as early as 10 weeks.)

Social Worker

(Some patients with social difficulties will benefit from early referral to a social worker.)

Graph type	Self Limiting Disease
Time 1	6 - 8 weeks
Time 2	38 - 42 weeks
Sex Ratios (M:F)	0:1
Age Range	15 - 45
Severity	Minor
Prevalence	38 in 1000 patients

LUMBOSACRAL STRAIN

(Low Back Injury, Low Back Pain, Back Injury, Bad Back, Back Sprain)

Disease Group: Injury and Other Miscellaneous Conditions

Symptoms:

*Severe pain in the lower back today
Fell downstairs yesterday
Can walk only with stick today

Clinical Signs/Examinations:

Moderately distressed
Limitation of flexion of the lumbar spine by 50 degrees
Tenderness of the lumbar spine

Laboratory Tests:

(no labtests present)

Management Plan:

<u>Optional:</u>

Compound Analgesic
(Often the pain requires a moderate analgesia. Rapid pain relief is essential.)

Benzodiazepine Sedative
(The pain in muscular injuries is often caused as much by muscualr spasm as by anything. A benzodiazepine for 5-10 days can be the most effective treatment. It also prevents most patients from rising from their bed because of the sedation and thus encourages bed rest!)

Osteopath
(Many patients will gain significant relief froman osteopath, although no direct referral is required, consent of the doctor may often be asked.)

<u>Unnecessary:</u>

Plain X-Ray, Lumbar Spine
(Usually a simple lumbra sprain requires no further investigation, UNLESS there is additional evidence of a prolapsed Intervertebral disc.)

Graph type	Self Limiting Disease
Time 1	12 - 48 hours
Time 2	10 - 20 days
Sex Ratios (M:F)	2:1
Age Range	25 - 55
Severity	Moderate
Prevalence	10 in 1000 patients

THYROTOXICOSIS

(Graves Disease, Toxic Goitre, Hyperthyroidism)

Disease Group: Allergic, Endocrine, Metabolioc and Nutritional Diseases

Symptoms:

*Sleeps lightly for the last 3 months
Rapid weight loss for the last 6 weeks
Continual palpitations for the last 2 weeks
Frequent moderate soreness in the left eye for the last 3 weeks
Soft swelling of the middle of the neck for the last 2 months
Has irregular periods for the last 6 months

Clinical Signs/Examinations:

Quite restless
Slightly cachectic
Swelling of the thyroid gland
Regular pulse rate of 120
Smooth swelling of the thyroid gland

Laboratory Tests:

Blood For Enzymes - TSH: 0.1,0.8
Blood For Enzymes - Total T4: 160,300

Management Plan:

Mandatory:

Thyroid Function Studies

(It is always essential to exclude thyrotoxicosis where there are supporting physical signs. Marginal tests are not conclusive, but a series of marginal tests marks a decrease in likelihood of thyrotoxicosis.)

Optional:

Beta-Blocker

(A Beta blocking drug may well offer the best treatment of the symptoms, and will certainly not harm a patient with simple anxiety only. An interim treatment until the diagnosis is confirmed and anti-thyroid treatment commenced.)

Anti Thyroid Drugs

(A possible treatment option once thyrotoxicosis has been confirmed. Usually a management option for an endocrinologist rather than a GP, at least on first presentation.)

Endocrinologist

(A hospital specialist is in the best position to decide, initially, on anti-thyroid treatment.)

General Physician

(A hospital specialist is in the best position, initially, to decide on anti-thyroid treatment.)

Graph type Disease Continues To Worsen With Time
Time 1 12 - 24 months
Time 2 (Not Applicable)
Sex Ratios (M:F) 1:8
Age Range 25 - 65

Severity	Fatal
Prevalence	1.4 in 1000 patients

DIABETES MELLITUS MATURITY ONSET

(Maturity Onset Diabetes Mellitus, Sugar Diabetes, Diabetes, Diabetes Mellitus, Maturity Onset Diabetes)

Disease Group: Allergic, Endocrine, Metabolioc and Nutritional Diseases

Symptoms:

*Some weight loss for the last 3 months
Ulcer on the top of the foot for the last 4 weeks
Always has to pass urine every 2 hours

Clinical Signs/Examinations:

Slightly cachectic
Lump on the foot
Tenderness of the foot
Pitting oedema of the foot

Laboratory Tests:

Urinalysis - Glucose: + + +

Management Plan:

Mandatory:

Advice

(The provision of a diabetic diet is mandatory for these patients and may control the condition without recourse to medication.)

Optional:

Oral Hypoglycaemics

(Probable first line treatment in maturity onset diabetes, after failure of diet alone.)

Graph type	Disease Continues To Worsen With Time
Time 1	2 - 10 months
Time 2	(Not Applicable)
Sex Ratios (M:F)	1:2
Age Range	45 - 65
Severity	Severe
Prevalence	23 in 1000 patients

ACNE VULGARIS
(Acne)

Disease Group: Diseases of the Skin and Cellular Tissues

Symptoms:

*Skin rash of the face for the last 3 months
*Skin rash of the upper back for the last 2 months
*Skin rash of the chest for the last 2 months

Clinical Signs/Examinations:

Pustular papular rash of the face
Pustular papular rash of the chest
Pustular papular rash of the upper back

Laboratory Tests:

(no labtests present)

Management Plan:

Optional:

Acne

(Topical treatments with scaling agents can help, as can erythema and scaling from Ultra Violet Light exposure. Retinoids applied topically also induce scaling. Keratolytics (benoyl Peroxide) can be very effective as can long term, low dosage antiobiotic treatment. Dermabrasion by abrasive pad or powder is particularly effective for cases where comedone predominate.)

Tetracycline

(Low dose tetracyclines for a prolonged period are often very effective, particularly in variants with predominant pustule formation.)

Erythromycin

(Where tetracyclines are ineffective, erythromycin can be a useful variant for pustule predominant cases.)

Graph type	Disease Remains At Constant State
Time 1	1 - 3 months
Time 2	(Not Applicable)
Sex Ratios (M:F)	1:1
Age Range	12 - 25
Severity	Minor
Prevalence	8.8 in 1000 patients

SENILE DEMENTIA

(No synonyms)

Disease Group: Mental,Psychoneurotic And Personality Disorders

Symptoms:

*Disorientated in place and time for the last 6 months
Can walk only with frame for the last 6 weeks
Had a fall outside yesterday

Clinical Signs/Examinations:

Limitation of cognitive function
Slightly distressed

Laboratory Tests:

Blood For Haematology - Haemoglobin: 8.0,14.0

Management Plan:

Mandatory:

Refer to Community Nurse
(May need general nursing care for poor hygiene Community Psychiatric Nurse may give support.)

Optional:

Patient to Return in 1 Months
(Continued contact is helpful to assess progress or deterioration.)

Graph type	Disease Continues To Worsen With Time
Time 1	1 - 20 years
Time 2	(Not applicable)
Sex Ratios (M:F)	1:1
Age Range	65 - 95
Severity	Severe
Prevalence	1.7 in 1000 patients

APPENDIX C

TUTORIAL RULES

This appendix contains a list of all the tutorial rules used in *P.C.Tutor*. This list is divided into three sublists covering Common Sense rules, Clinical Examinations rules, and Problem-Solving rules. Note that an underscore character '_' shown in the rules means that any value associated with the particular parameter replaced with '_' is ignored by the rule.

C.1 COMMON SENSE RULES

The following list of rules show the Common Sense rules currently applied by the system during the case Management Plan Formulation stage. These rules are based on the 30 diseases and problems in the medical database. It is expected that this list will be extended further when new diseases and problems are added to the database.

Rule Format:

sense(Age, Sex, Management_Action) if <condition>.

Prolog Format of Rules:

R1: sense(Age, Sex, sign_contraceptive_advice_form) if
Sex = female and
Age > = 35.

R2: sense(_, Sex, Request_Ultrasound_Foetal_Heart_test) if
Sex = female.

R3: sense(_, Sex, Request_Full_Obstetric_Ultrasound_test) if
Sex = female.

R4: sense(_, Sex, Request_Cervical_Cytology_test) if
Sex = female.

R5: sense(_, Sex, Request_Semen_Analysis_test) if
Sex = male.

R6: sense(_, Sex, Refer_to_Obstetrician) if
Sex = female.

R7: sense(_, Sex, Refer_to_Gynaecologist) if
Sex = female.

R8: sense(Age, _, Refer_to_Geriatrician) if
Age >= 65 years.

R9: sense(Age, _, Refer_to_Paediatrician) if
Age <= 15 years

R10: sense(_, Sex, Give_Male_Hormone_Antagonist) if
Sex = male.

R11: sense(_, Sex, Give_Vaginal_Anti_Bacterial_Agents) if
Sex = female.

R12: sense(_, Sex, Give_Combined_Oral_Contraceptive) if
Sex = female.

R13: sense(_, Sex, Give_Progesterone_Only_Contraceptive) if
Sex = female.

R14: sense(_, Sex, Give_Spermicide) if
Sex = female.

R15: sense(_, Sex, Request_Pregnancy_test) if
Sex = female.

R16: sense(_, Sex, Request_High_Vaginal_Swab_test) if
Sex = female.

C.2 CLINICAL EXAMINATIONS RULES

The Examinations rules executed during the Examinations phase of the data gathering stage are based on the 19 disease categories specified by Hodgkin [1985]. These are:

1. Infective and Parasitic Diseases
2. Neoplasms
3. Allergic, Endocrine, Metabolic and Nutritional Diseases
4. Diseases of the Blood and Blood-forming Organs
5. Mental, Psychoneurotic and Personality Disorders
6. Diseases of the Nervous System
7. Diseases of the Eye
8. Diseases of the Ear and Mastoid Processes
9. Diseases of the Circulatory System
10. Diseases of the Respiratory System
11. Diseases of the Digestive System
12. Diseases of the Urinary System
13. Diseases of the Male Genital System
14. Diseases of the Female Genital System
15. Family Planning, Complications of Pregnancy, Childbirth & the Puerperium
16. Diseases of the Skin and Cellular Tissues
17. Diseases of the Bones and Organs of Movement
18. Congenital Malformations and Certain Problems of Early Infancy
19. Injury and Other Miscellaneous Problems

The examinations rules are divided into two groups: (1) Those associated with the Disease Categories, and (2) Those associated with the Primary Sites and/or types of symptoms. The rules in these groups will be shown in their English format.

The rules in this section have been revised on a number of times. However, it is expected that further revisions will be required when more diseases/problems are added to the database.

C.2.1 Rules Associated with Disease Categories

Note also that the group numbers shown in the rules are based on the disease groups above.

R1:
IF 1) disease group 1, and
2) the requested examination is a member of the following list:
General Condition, or
URT, or
LRT, or
Cardiovascular System, or
Central Nervous System, or
Reticulo-Endothelial System, or
Abdomen, or
Chest & Thorax, or
Genito-urinary System, or
Genital Tract, or
Endocrine System, or
Musculoskeletal System, or
Skin, or
Gastrointestinal Tract, or
Auscultation, or
Percussion
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R2:
IF 1) disease group 2, and
2) the requested examination is a member of the following list:
General Condition, or
URT, or
LRT, or
Cardiovascular System, or
Abdomen, or
Chest & Thorax, or
Musculoskeletal System, or
Skin, or
Percussion
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R3:

IF 1) disease group 3, and
2) the requested examination is a member of the following list:
General Condition, or
URT, or
LRT, or
Cardiovascular System, or
Reticulo-Endothelial System, or
Abdomen, or
Skin
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R4:

IF 1) disease group 4 and
2) the requested examination is a member of the following list:
General Condition, or
URT, or
Cardiovascular System, or
Reticulo-Endothelial System, or
Abdomen, or
Chest & Thorax
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R5:

IF 1) disease group 5, and
2) the requested examination is a member of the following list:
General Condition, or
Central Nervous System
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R6:

IF 1) disease group 6, and
2) the requested examination is a member of the following list:
General Condition, or
Central Nervous System, or
Reticulo-Endothelial System, or
Musculoskeletal System
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R7:

IF 1) disease group 7, and
2) the requested examination is a member of the following list:
General Condition, or
URT
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R8:

IF 1) disease group 8, and
2) the requested examination is a member of the following list:
General Condition, or
Central Nervous System
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R9:

IF 1) disease group 9, and
2) the requested examination is a member of the following list:
General Condition, or
Cardiovascular System, or
Reticulo-Endothelial System
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R10:

IF 1) disease group 10, and
2) the requested examination is a member of the following list:
General Condition, or
URT, or
LRT, or
Reticulo-Endothelial System, or
Chest & Thorax
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R11:
IF 1) disease group 11, and
2) the requested examination is a member of the following list:
General Condition, or
URT, or
Reticulo-Endothelial System, or
Abdomen, or
Genital Tract, or
Gastrointestinal Tract
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R12:
IF 1) disease group 12, and
2) the requested examination is a member of the following list:
General Condition, or
Reticulo-Endothelial System, or
Genito-urinary System, or
Genital Tract
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R13:
IF 1) disease group 13, and
2) the requested examination is a member of the following list:
General Condition, or
Reticulo-Endothelial System, or
Genito-urinary System, or
Genital Tract
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R14:
IF 1) disease group 14, and
2) the requested examination is a member of the following list:
General Condition, or
Reticulo-Endothelial System, or
Genito-urinary System, or
Genital Tract
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R15:
IF 1) disease group 15, and
2) the requested examination is a member of the following list:
General Condition, or
Abdomen
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R16:
IF 1) disease group 16 and
2) the requested examination is a member of the following list:
General Condition, or
Skin
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R17:
IF 1) disease group 17, and
2) the requested examination is a member of the following list:
General Condition, or
Musculoskeletal System
THEN report the results of the examination
ELSE
1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R18:
IF 1) disease group 18, and
2) the requested examination is a member of the following list:
General Condition, or
Musculoskeletal System, or
Skin
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

R19:
IF 1) disease group 19, and
2) the requested examination is a member of the following list:
General Condition, or
Musculoskeletal System
THEN report the results of the examination
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF

C.2.2 Rules Associated with Primary Sites and/or Types of Symptoms

```
R1:
IF      the primary site/type of symptom is Hearing
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Central Nervous System, or
                        URT
        THEN  report the examination results
        ELSE  1) display "inappropriate examination" error message, and
              2) update student model
        ENDIF
ENDIF

R2:
IF      the primary site/type of symptom is Smell, Taste, or Appetite
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Central Nervous System, or
                        URT, or
                        Gastrointestinal Tract
        THEN  report the examination results
        ELSE  1) display "inappropriate examination" error message, and
              2) update student model
        ENDIF
ENDIF

R3:
IF      the primary site/type of symptom is Vision
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Central Nervous System
        THEN  report the examination results
        ELSE  1) display "inappropriate examination" error message, and
              2) update student model
        ENDIF
ENDIF
```

R4:

```
IF      the primary site/type of symptom is Locomotor System
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Musculoskeletal System
        THEN    report the examination results
        ELSE    1) display "inappropriate examination" error message, and
                2) update student model
        ENDIF
ENDIF
```

R5:

```
IF      the primary site/type of symptom is Central Nervous System
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Central Nervous System
        THEN    report the examination results
        ELSE    1) display "inappropriate examination" error message, and
                2) update student model
        ENDIF
ENDIF
```

R6:

```
IF      the primary site/type of symptom is Respiratory System
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        URT, or
                        LRT, or
                        Reticulo-Endothelial System, or
                        Chest & Thorax, or
                        Auscultation, or
                        Percussion
        THEN    report the examination results
        ELSE    1) display "inappropriate examination" error message, and
                2) update student model
        ENDIF
ENDIF
```

R7:

```
IF      the primary site/type of symptom is Cardiovascular System
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Cardiovascular System, or
                        Reticulo-Endothelial System
        THEN    report the examination results
        ELSE    1) display "inappropriate examination" error message, and
                2) update student model
        ENDIF
ENDIF
```

R8:

```
IF      the primary site/type of symptom is Genito-urinary System
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Abdomen, or
                        Reticulo-Endothelial System, or
                        Genito-urinary System, or
                        Genital Tract
        THEN    report the examination results
        ELSE    1) display "inappropriate examination" error message, and
                2) update student model
        ENDIF
ENDIF
```

R9:

```
IF      the primary site/type of symptom is Gastrointestinal System
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Abdomen, or
                        Reticulo-Endothelial System, or
                        Gastrointestinal Tract
        THEN    report the examination results
        ELSE    1) display "inappropriate examination" error message, and
                2) update student model
        ENDIF
ENDIF
```

R10:

```
IF      the primary site/type of symptom is Sweating, Shivering, Sneezing, Coldness
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Skin
        THEN    report the examination results
        ELSE    1) display "inappropriate examination" error message, and
                2) update student model
        ENDIF
ENDIF
```

R11:

```
IF      the primary site/type of symptom is Bruising, Colouration, Skin Rash
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Reticulo-Endothelial System, or
                        Skin
        THEN    report the examination results
        ELSE    1) display "inappropriate examination" error message, and
                2) update student model
        ENDIF
ENDIF
```

R12:

```
IF      the primary site/type of symptom is Lumps, Swelling
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Reticulo-Endothelial System, or
                        Skin
        THEN    report the examination results
        ELSE    1) display "inappropriate examination" error message, and
                2) update student model
        ENDIF
ENDIF
```

R13:

```
IF      the primary site/type of symptom is Cuts, Grazes, Ulcers, Bleeding
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Reticulo-Endothelial System, or
                        Skin
        THEN    report the examination results
        ELSE    1) display "inappropriate examination" error message, and
                2) update student model
        ENDIF
ENDIF
```

R14:

```
IF      the primary site/type of symptom is Weight Loss
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        Reticulo-Endothelial System
        THEN    report the examination results
        ELSE    1) display "inappropriate examination" error message, and
                2) update student model
        ENDIF
ENDIF
```

R15:

```
IF      the primary site/type of symptom is Aches, Soreness, Pain
THEN
        IF      the requested examination is a member of the following list:
                        General Condition, or
                        URT, or
                        Central Nervous System, or
                        Abdomen, or
                        Chest & Thorax, or
                        Musculoskeletal System
        THEN    report the examination results
        ELSE    1) display "inappropriate examination" error message, and
                2) update student model
        ENDIF
ENDIF
```

R16:
IF the primary site/type of symptom is Itching, Numbness, Pins & Needles, Tingling, Stiffness
THEN
IF the requested examination is a member of the following list:
General Condition, or
URT, or
Central Nervous System, or
Abdomen, or
Chest & Thorax, or
Musculoskeletal System, or
Skin
THEN report the examination results
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF
ENDIF

R17:
IF the primary site/type of symptom is Accident
THEN
IF the requested examination is a member of the following list:
General Condition, or
Musculoskeletal System, or
Skin
THEN report the examination results
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF
ENDIF

R18:
IF the primary site/type of symptom is Family Problem, or Marital Problem, or Illness, or Work Problem, or Emotional Problem, or Drug Abuse
THEN
IF the requested examination is a member of the following list:
General Condition
THEN report the examination results
ELSE 1) display "inappropriate examination" error message, and
2) update student model
ENDIF
ENDIF

C.3 PROBLEM-SOLVING RULES

The Problem-Solving rules have been implemented as part of the *P.C.Tutor* source code. For this reason, the following list of rules are shown in their English equivalence and in high level format. This set of rules is based on basic problems' detection and elimination processes. Specific medical diagnostic rules appropriate to each disease/problem in the medical database will be included in the next version of *P.C.Tutor* (i.e. in SIMPLE LINCTUS - see Chapter 7 section 7.6). The Problem-Solving rules are executed during Hypothesis entry/refinement and during the disease/problem diagnosis stage.

R1: The elimination of a disease/problem by the existence of an abnormal set of data.

```
IF      the results of the requested set of data are abnormal
THEN    mark as deleted all the diseases/problems in the tutor's differential diagnosis where
the results
        of the same set of data are normal
ENDIF
```

R2: The elimination of a disease/problem by the absence of abnormal data.

```
IF      the results of the requested set of data are normal
THEN    mark as deleted all the diseases/problems in the tutor's differential diagnosis where
        the results of the same set of data are abnormal
ENDIF
```

R3: Formulating an initial or updating the current hypothesis list

```
get student action
IF      student action = add-new
THEN    1) get disease/problem name
        2)
        IF      the disease/problem is not part of the database
        THEN    Label:  1) display disease/problems categories list, and
                        2) ask student to select category
                        IF      category is empty
                        THEN    display message "Category is empty - select another group
                                or ignore"
                                get student reply
                                IF      reply = select another
                                THEN
                                        REPEAT
                                        1) display disease/problems categories list, and
                                        2) ask student to select category
                                        IF      category is empty
                                        THEN    display message "Category is empty -
                                                select another group or ignore"
                                                get student reply
                                        ENDIF
                                        UNTIL the selected diseases/problems category is
                                                not empty
                                        display all diseases/problems in category
                                        ask student to select a disease/problem or ignore
                                        get student reply
                                        IF      reply < > ignore
                                        THEN    append selected disease/problem to
                                                student hypothesis
```

```
                        ENDIF
                    ENDIF
                ELSE 1) display all diseases/problems in category
                     2) ask student to select a disease/problem or ignore
                     3) get student reply
                     IF      reply < > ignore
                     THEN    append selected disease/problem to student
                             hypothesis
                     ENDIF
                ENDIF
            ENDIF
        ELSE IF     1) the new disease/problem is not a member of the tutor's
                           differential diagnosis, and
                    2) the disease/problem is part of the database
             THEN   1) generate a multiple choice question using all case data {see
                           Chapter 5 section 5.2.2.3 for question format details}, and
                    2) update student model
             ELSE   1) acknowledge correct disease/problem, and
                    2) update student model
             ENDIF
        ENDIF
ELSE
        IF      student action = delete-problem
        THEN    1) get disease/problem name
                2)
                IF      selected disease/problem has been marked as deleted in the tutor's
                        differential diagnosis
                THEN
                        IF      student model is above average
                        THEN    acknowledge correct elimination
                        ELSE    acknowledge correct elimination stating the facts that led
                                to such conclusion
                        ENDIF
                        update student model
                ELSE    1) generate a multiple choice question using all case data {see
                                Chapter 5 section 5.2.2.3 for question format details}, and
                        2) update student model
                        3) undelete disease/problem from student hypothesis
                ENDIF
        ENDIF
ENDIF
```

R4: Helping the student in formulating a hypothesis

```
IF      1) current process is Hypothesis, and
        2) the student has requested HELP
THEN
        IF      there are a number of diseases/problems in the tutor's differential diagnosis
                that were not stated by the student
        THEN
                LOOP:
                        IF      one of these diseases/problems was considered by the
                                student during a previous consultation
                        THEN    1) offer disease/problem to student for consideration, and
                                2) append disease/problem to student hypothesis
                        ELSE    1) state one disease/problem from the tutor's differential
                                        diagnosis, and
```

2) append disease/problem to student hypothesis
ENDIF
update student model
UNTIL (number of given diseases/problems = 2) OR (no more diseases/problems were found in the tutor's differential diagnosis)
ELSE display "can't give you any help" message
ENDIF
ENDIF

R5: Giving the student help during the Examinations stage
IF 1) current process is Examinations, and
2) the student has requested HELP
THEN
IF student hypothesis is empty
THEN display all possible examinations based on the primary sites/types of symptoms
ELSE display all possible examinations based on the categories of the diseases/problems stated in the student's hypothesis
ENDIF
update student model
ENDIF

R6: Giving the student help during the Laboratory Tests stage (This rule apply to tests made in the GP surgery
IF 1) current process is Laboratory Tests, and
2) the student has requested HELP, and
3) student hypothesis is not empty
THEN
IF 1) any of the diseases/problems in the student hypothesis has an associated test, and
2) the test has not been requested by the student
THEN 1) display test name
2) update student model
ELSE display "can't help you" message
ENDIF
ENDIF

R7: Moving to the Management Plan formulation stage with a serious disease/problem missing from the student hypothesis
IF 1) current process is Management-Plan-Formulation, and
2) there is at least one serious disease/problem considered by the tutor in its differential diagnosis and
3) the serious disease/problem is missing from the student's differential diagnosis, and
4) all the abnormal case data have been displayed
THEN 1) question the student of the serious disease/problem by asking him/her to select the appropriate case data that are applicable to this disease/problem
2) update student model according to selection, and
3) append the disease/problem to the student hypothesis, and
4) ask the student to consider this disease/problem when managing the case
ENDIF

R8: Stating the final diagnosis

```
IF      main disease/problem is the only one in the student hypothesis
THEN    1) acknowledge correct diagnosis, and
        2) update student model
ELSE
        IF      main disease/problem and one other or more diseases/problems are present
                in the student hypothesis
        THEN
                LABEL:
                1) get disease/problem diagnosis
                2)
                IF      diagnosis is correct
                THEN    1) acknowledge correct diagnosis
                        2) update student model
                ELSE    1) state correct diagnosis
                        2) update student model
                        3)
                        IF      student level is not high
                        THEN    state how the other diseases/problems in the hypothesis
                                could have been eliminated
                        ENDIF
                ENDIF
        ELSE
                IF      main disease/problem is missing from student hypothesis
                THEN
                        IF      one of the diseases/problems in the student hypothesis has
                                been marked as deleted by the tutor
                        THEN    1) ask, by displaying all the facts that resulted in the
                                        elimination of the disease/problem, the student to
                                        select the facts that justify the existence of the
                                        disease/problem
                                2) reply to student selection, and
                                3) update student model
                        ELSE    1) update student model
                                2) goto LABEL:
                        ENDIF
                ENDIF
        ENDIF
ENDIF
```

APPENDIX D

P.C.TUTOR SESSION CONTROL PROCEDURES

The following is a top level Prolog pseudo-code version of the main control procedures in *P.C.Tutor.*

LEVEL 0 - TOP SYSTEM LEVEL

```
main:-
        display_front_page,       /* This predicate displays the system title (i.e. P.C.Tutor or
                                  SIMPLE LINCTUS in the extended version) */

        logon(USER_ACTION), % see Level 1
                not(USER_ACTION='quit'),

        open_all_databases,       % open all disk-based databases for access

        repeat,
                get_case,                       % see Level 1
                start_tutoring_environment,     % see Level 1
                end_of_case,                    % see Level 1
                revision,                       % see Level 1
        system_flag = 'quit',!,

        close_databases.                        % closes all disk-based databases
main.
```

LEVEL 1 - (LOGON)

```
logon(USER_ACTION):-
        /*
        get user name and password. The user at this stage can stop the system by pressing
        the ABORT key.
        */
        get_user_name_and_password(USER_ACTION),

        /*
        if the ABORT key was not pressed, the system checks for any previously aborted
        cases. If any were found, the user is told of their types. There are only two types:
                1. Cases aborted during the data gathering process.
                2. Cases waiting for laboratory tests results.
        */
        check_for_outstanding_cases(USER_ACTION),
        display_system_instructions.        % depending on the users choice
```

LEVEL 1 - (GET-CASE)

```
get_case:-
        open_student_model_database,

        /*
        retrieve any cases waiting for laboratory tests.
        if      none were found
        then    continue
        else    complete all cases before starting any new case.

        Completion of cases is done by jumping to the management processing predicates.
        This made by setting the system_flag to 'manageit'.
        */
        retrieve_unfinished_cases,

        /*
        retrieve a case that was aborted prior to managing it
        if      no case was found
        then    continue
        else    complete the case before starting a new one
        */
        retrieve_aborted_case,

        /*
        The following predicates will not be executed if there any any outstanding cases are
        present from last session
        */
        get_number_of_cases_for_this_session,   % if number=0, system_flag is
                                                % set to 'quit'

        start_new_case.  % if system_flag = 'quit' this will not be executed
```

LEVEL 1 - (START-TUTORING-ENVIRONMENT)

```
start_tutoring_environment:-system_flag = 'quit',!.
start_tutoring_environment:-system_flag = 'manageit',!.
start_tutoring_environment:-
        display_aborted_case,     % if exists

        % begin main tutoring environment loop
        repeat,
                next_menu_option,       % see Level 2

        % set the approporiate system_flag value
        case_completed,

        system_flag = 'quit' or 'manageit'.
```

LEVEL 1 - (END-OF-CASE)

```
end_of_case:-system_flag = 'quit',!.
end_of_case:-
        /*
        This predicate will be executed if the current case is not an outstanding case. Check
        for any requests for laboratory tests in the student's management plan. If any were
        found, set a special flag.
        */
        check_for_lab_tests,

        /*
        This predicate will execute other lower level predicates to perform the following set of
        actions:
                1. Display the management plan response
                2. Display any case data that were not displayed to the student
                3. Depending on the student level, ask for case diagnosis (if level < > high)
                4. Inform the student how the other problems (if any) in his/her hypothesis
                        can be eliminated
        */
        process_unfinished_case,


        % This predicate will not be executed if the above predicate was true
        process_current_management_plan,

        /*
        The following predicate will be executed if any laboratory tests were detected by
        check_for_lab_tests predicate above. The flag set in the check_for_lab_tests is tested
        by this predicate.
        */
        stop_case,

        /*
        This predicate will be executed if the current case is not an outstanding case

        This predicate will execute other lower level predicates to perform the following set of
        actions:
                1. Display the management plan response
                2. Display any case data that were not displayed to the student
                3. Depending on the student level, ask for case diagnosis (if level < > high)
                4. Inform the student how the other problems (if any) in his/her
                        hypothesis can be eliminated
        */
        end_of_case_management.
```

LEVEL 1 - (REVISION)

```
revision:-system_flag = 'quit',!.
revision:-
        /*
        The options in the End of Case Menu are:
                a. Print current case log
                b. Display current case log
                c. Save current case log to permanent file
                d. Review a particular problem
                e. Cross reference problems by management actions
                f. Cross reference problems by symptoms
                g. Display student level graph
                h. Print consultations report on present medical condition
        */
        display_and_process_end_of_case_menu,

        process_end_of_current_case.     % see Level 2
```

LEVEL 2 - (NEXT-MENU-OPTION)

```
next_menu_option:-
        new_case_flag = 'false',

        % The log will be read and used as the current log
        read_previous_log,

        /*
        The Main Tutoring Menu includes the following options:
                1. Patient Record
                2. Symptoms
                3. Examinations
                4. Tests
                5. Hypothesis
                6. Management
                7. View Current Log
                8. Review Case Data Separately
                9. Print current log
        */
```

{The display_and_process_main_menu procedure is executed as a repeat...until loop. The user can start any of the procedures listed above by either pressing a special function key or moving the highlight over the option and pressing the Enter key. The loop is completed when either of the following states are achieved:

1. The user aborts the consultation during the data gathering stage; or

2. The user has started the Management Plan Formulation stage.

The tutorial rules used by the system (see Appendix C) are executed during the following stages:

1. Common Sense rules - during Case Management

2. Examinations rules - during Examinations

3. Problem solving rules - during Hypothesis entry/refinement and Case Diagnosis after the Managment Plan's tutor feedback}

```
        display_and_process_main_menu,!.

next_menu_option:-display_and_process_main_menu.
```

LEVEL 2 (PROCESS-END-OF-CURRENT-CASE)

```
process_end_of_current_case:-
        last_case_flag = 'true', set_system_flag_to_quit,!.
process_end_of_current_case:-
        set_system_flag_to_continue.     % continue the main loop in Level 0
```

APPENDIX E

AN EXAMPLE P.C.TUTOR SESSION

The example session shows most of the features in P.C.Tutor. However, I must emphasize that the screen display is different from the printed version. The screen display uses pull-down menus, pop-up menus, function key selection, highlighting of text, multiple windows, and other IBM PC display features.

In the example below, all user responses to *P.C.Tutor* are shown in **BOLDFACE** and the annotations showing the system's procedure execution are enclosed between the braces ({}). Subject starting points are indicated by (−−»). These points are shown here for the purpose of clarifying the system's work.

PRIMARY CARE TUTOR

UNIVERSITY OF SHEFFIELD
Department of Computer Science
Department of General Practice

Thursday 1st March 1990

Enter your surname: **Mansour**
Enter your password: *********

Welcome to SIMPLE LINCTUS Dr. Mansour

−−» *System instructions*

> *{The user is asked if he wishes to view the system instructions each time he logs on}*

Do you want the instructions for using the system (y/n) ? N

--» *Tutorial Description*

{The tutorial description is shown to first time users only}

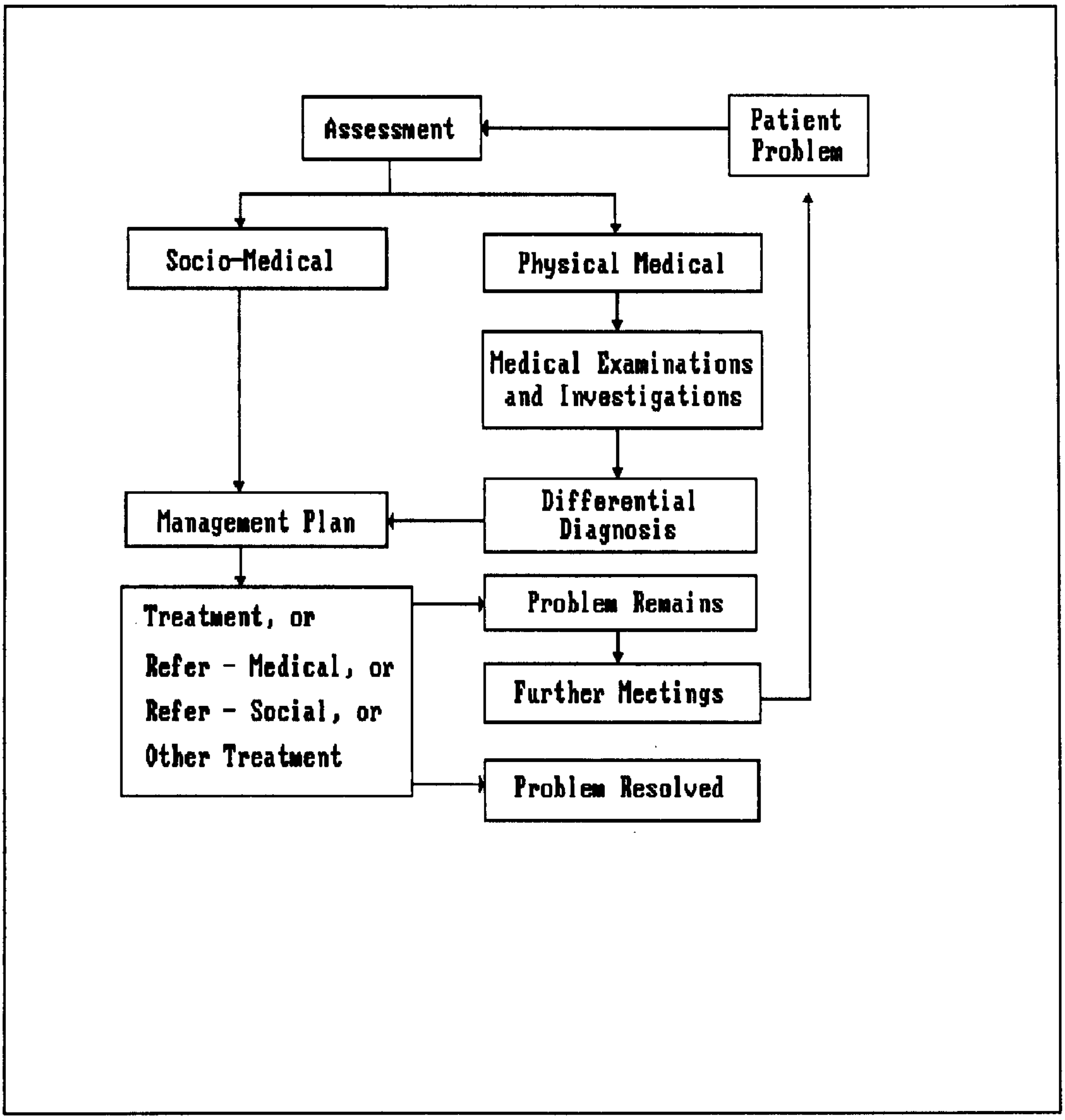
Brief description of the Tutorial Session

You are a General Practitioner in a Sheffield surgery. The surgery has the standard basic equipment for examinations and laboratory tests. During each session, you see up to ten patients. However, there may be some additional patients from your previous consultation whom you sent for external laboratory tests.

Each patient will present you with up to two presenting symptoms. At this stage, you will be given the choice of giving your initial hypothesis, which will be based on the patient record and the presenting symptom(s), or getting more information from the patient through questions and/or examinations.

You SHOULD always let the tutor know of the problem(s) you are thinking about by entering such problem(s) in the hypothesis list. This will allow the tutor to assess your progress and guide you accordingly.

The tutor operates by following the general Medical Consultation Model shown below and in particular the Physical Medical route:



How many patients would you like to see today ? **1**
(Maximum number of patients per session is 10. Enter 0 to Abort)

--» *Patient Selection*

{A patient is selected according to the population and consultation profiles}

Selecting a new patient. Please wait ...

Name: A. Jones
Age: 21 year(s) and 8 month(s)
Sex: Female
Address:
232 Totley Rise
Doncaster
South Yorkshire

--» *Problem Selection*

{Once the patient has been selected a problem is selected that is appropriate to the age and sex of the patient. A maximum of two presenting symptoms is given to the user at this stage}

A. Jones is complaining of the following symptom(s):

* Moderate sore throat for the last 3 days

--» *Tutor's Differential Diagnosis Generation*

{At this stage, the tutor generates its own differential diagnosis. This will be based on the patient's age, sex, and presenting symptom(s). The differential diagnosis in this case includes Glandular Fever and Viral Pharyngitis}

> > Generating the differential diagnosis for this case. Please wait ...

Having considered the patient record and presenting symptom(s), do you wish to:

1. Enter your initial hypothesis (i.e. differential diagnosis).
2. Request more case data.

Enter your choice (1 or 2) ? **2**

{The user has opted to request further case data before making an initial differential diagnosis}

--------- Main Tutoring Menu ---------
A. Patient record
B. Symptoms
C. Examinations
D. Tests
E. Hypothesis
F. Management
G. View Log
H. Review case data separately
I. Print current log

Select an option: **B (Symptoms)**

--------- Symptoms Main Menu ---------
» Next important presenting symptom «
A. Altered Sensorium
B. Physical Dysfunction
C. Self-Perceived Signs
D. Physical & Life Events

{The first option in the Symptoms Main Menu is an undirected question which results in displaying any symptoms specified by the expert GP as presenting (i.e. has an '' when specified in the medical database}*

Select an option: **(Next important presenting symptom)**
> > Sorry, no more presenting symptoms to report on

Select an option: **A (Altered Sensorium)**

--------- Altered Sensorium Menu ---------
Hearing
Smell, Taste, Appetite
Vision
Impaired Consciousness
Vertigo and Dizziness
Aches, Soreness, Pain
Itching, Numbness, Pins & Needles, Tingling, Stiffness

Select an option: **(Aches, Soreness, Pain)**
** dull aching in the lower back for the last 5 days

Select an option: **(Itching, Numbness, Pins & Needles, Tingling, Stiffness)**
** Itching, Numbness, Pins & Needles, Tingling, Stiffness: Normal

{In the current version of P.C.Tutor, a user is not marked down if he/she requests symptoms which are normal. Request relevance is not taken into account in this version}

Select an option: {student presses the Esc key to return to the Symptoms Main Menu}

--------- Symptoms Main Menu ---------
» Next important presenting symptom «
A. Altered Sensorium
B. Physical Dysfunction
C. Self-Perceived Signs
D. Physical & Life Events

Select an optio: **B (Physical Dysfunction)**

--------- Physical Dysfunction Menu ---------
Locomotor System
Central Nervous System
Respiratory System
Cardiovascular System
Genito-Urinary System
Gastro-Intestinal System

Select an option: **(Respiratory System)**
** (Respiratory System): Normal

Select an option: {student presses the Esc key to return to the Symptoms Main Menu}

Select an option: C **(Self-Perceived Signs)**

--------- Self-Perceived Signs Menu ---------
Sweating, Shivering, Sneezing, Coldness
Bruising, Colouration, Skin rash
Lumps, Swelling
Cuts, Grazes, Ulcers, Bleeding
Weight Loss

Select an option: **(Sweating, Shivering, Sneezing, Coldness)**
** sweating

Select an option: **(Bruising, Colouration, Skin Rash)**
** Bruising, Colouration, Skin rash: Normal

{The user was thinking about Rubella when he asked about any skin rash. Rubella is sometimes considered in a case like this (Hodgkin [1985]). But having found it to be normal, the user has discounted it}

Select an option: {student presses the Esc key to return to the Symptoms Main Menu}

--------- Symptoms Main Menu ---------
» Next important presenting symptom «
A. Altered Sensorium
B. Physical Dysfunction
C. Self-Perceived Signs
D. Physical & Life Events

Select an option: {student presses the Esc key to return to the Main Tutoring Menu}

--------- Main Tutoring Menu ---------
A. Patient record
B. Symptoms
C. Examinations
D. Tests
E. Hypothesis
F. Management
G. View Log
H. Review case data separately
I. Print current log

Select an option: C **(Examinations)**

--------- Examinations Main Menu ---------
A. General Condition
B. Examine Body Area
C. Auscultation
D. Percussion

Select an option: A **(General Condition)**
** moderately distressed
** very pyrexial

Select an option: **B (Examine Body Area)**

--------- Body Areas Main Menu ---------

A. Upper Respiratory Tract
B. Lower Respiratory Tract
C. Cardiovascular System
D. Central Nervous System
E. Reticulo-Endothelial System
F. Abdomen
G. Chest and Thorax
H. Genito-Urinary System
I. Genital Tract
J. Endocrine System
K. Musculoskeletal System
L. Skin
M. Gastrointestinal Tract

Select an option: **A (URT)**
Select action - (I)nspect, (P)alpate: **I**
(Upper Respiratory Tract on inspection): redness of the soft palate

Select an option: {student presses the Esc key to return to the Examinations Main Menu}

--------- Examinations Main Menu ---------

A. General Condition
B. Examine Body Area
C. Auscultation
D. Percussion

Select an option: {student presses the Esc key to return to the Main Tutoring Menu}

--------- Main Tutoring Menu ---------

A. Patient record
B. Symptoms
C. Examinations
D. Tests
E. Hypothesis
F. Management
G. View Log
H. Review case data separately
I. Print current log

Select an option: **H (Hypothesis)**

--» *Initial Differential Diagnosis*

{The user is now entering his initial differential diagnosis}

Having considered the patient record and other case data, please enter your list of problems which you wish to consider as your initial hypothesis:

** **sore throat**
** **flu**

{The user has entered the synonyms of the two diseases in place of their correct medical names as specified in the medical database}

--» *Incorrectly Stated Disease*

{The user is going to be questioned on the facts that suggested the possible existence of Influenza. In this case, Influenza was eliminated by the patient's age. According to the medical database, Influenza affects patients from 25 to 85 years old!! (see Appendix B for details of all the diseases/problems entered in the database by the GP working on the CTISS project). While the patient in this case is about 21 years old.

The questioning is done by showing all the case data displayed to him prior to entering the hypothesis and asking him to select the relevant data that confirmed the possible existence of the disease.}

From the following list of already known case facts, select the ones which indicate to you the possible existence of Influenza:

Patient Record & Symptoms List

1. age
2. sex
3. moderate sore throat for the last 3 days
4. Respiratory System - Normal
5. sweating
6. dull aching in the lower back for the last 5 days
7. Itching, Numbness, Pins & Needles, Tingling, Stiffness - Normal
8. Bruising, Colouration, Skin rash - Normal
9. none of the above

Enter your choice: **3,4,5,6**

Clinical Examinations List

10. (General Condition): moderately distressed
11. (General Condition): very pyrexial
12. (Upper Respiratory Tract on inspection): redness of the soft....
13. none of the above examinations

Enter your choice: **10,11**

Evaluation of your selected problem (Influenza)

The reason for asking you to justify the existence of the above problem is:

(a) the presenting symptoms of Influenza as entered in the system's database are not similar to those of the case that you are currently working on; and/or
(b) the patient age and/or sex in this case do not apply to Influenza.

- The above problem affects patients from 25 to 85 years old.

- There are similar abnormalities to the following symptoms in your selected problem and the one you are currently working on:

* moderate sore throat for the last 3 days
* sweating
* dull aching in the lower back for the last 5 days

- The following symptoms are normal in this case (i.e. the one you are working on). While in the above problem these symptoms are abnormal:

* Respiratory System

- There are similar abnormalities to the following signs in your selected problem and the one you are currently working on:

* (General Condition): moderately distressed
* (General Condition): very pyrexial

{In this case, Influenza was not considered by the tutor due to the different patient's age and the normal result obtained from the Respiratory System symptom's request. The student model (hypothesis formulation section) has been decreased by the equivalent of one disease/problem as a result of this entry. Diseases/problems in the tutor's differential diagnosis are represented as primary (i.e. those with similar number of presenting symptoms) and secondary (i.e. those with less number of common presenting symptoms). The hypothesis section credits are divided by the number of primary diseases/problems in the tutor's differential diagnosis}

--» *View Medical Data*

{The user is offered to view the details of the incorrecly stated disease available in the system's database}

Your choice of Influenza is incorrect.

Therefore, I will not consider it as part of your hypothesis.

In the meantime, would you like to see any details about Influenza? (y/n) N

The problems which you have entered as your differential diagnosis (listed below) are correct:

▪ Viral pharyngitis
(Incidence: 91.3 in 1000 patients visiting the surgery)

{The tutor has confirmed the existance of Viral Pharyngitis as part of its differential diagnosis. The Incidence of this disease is shown for reference purposes}

--------- Main Tutoring Menu ---------
A. Patient record
B. Symptoms
C. Examinations
D. Tests
E. Hypothesis
F. Management
G. View Log
H. Review case data separately
I. Print current log

Select an option: **E (Examinations)**

--------- Examinations Main Menu ---------
A. General Condition
B. Examine Body Area
C. Auscultation
D. Percussion

Select an option: **HELP**

--» *Requesting Help*

> *{The request for help has resulted in the student model being decreased. The amount of credits lost (5 credits - this is an arbitrary value) is fixed in this case. The help information given is based on the disease category of the disease currently being considered by the user. The examinations appropriate to the disease are listed to the student}*

Since the problem(s) in your hypothesis (listed below) belong to the following group(s):

Viral pharyngitis (Infective and Parasitic Diseases)

Therefore, based on the above group(s), the following examinations/examination areas are appropriate in this case:

General Condition
Upper Respiratory Tract
Lower Respiratory Tract
Central Nervous System
Cardiovascular System
Reticulo-Endothelial System
Chest and Thorax
Abdomen
Gastrointestinal Tract
Genito-Urinary System
Genital Tract
Endocrine System
Musculoskeletal System
Skin
Auscultation
Percussion

Select an option: **B (Examine Body Area)**

--------- Body Areas Main Menu ---------
- A. Upper Respiratory Tract
- B. Lower Respiratory Tract
- C. Cardiovascular System
- D. Central Nervous System
- E. Reticulo-Endothelial System
- F. Abdomen
- G. Chest and Thorax
- H. Genito-Urinary System
- I. Genital Tract
- J. Endocrine System
- K. Musculoskeletal System
- L. Skin
- M. Gastrointestinal Tract

Select an option: **E (Reticulo-Endothelial System)**
Select action - (I)nspect, (P)alpate: **P**

(Reticulo-Endothelial System on palpation): tender swelling of the cervical lymph nodes
(Reticulo-Endothelial System on palpation): tender swelling of the axillary lymph nodes
(Reticulo-Endothelial System on palpation): tender swelling of the submandibular lymph nodes
(Reticulo-Endothelial System on palpation): painless swelling of the spleen

Select an option: {student presses the Esc key to return to the Examinations Main Menu}

--------- Examinations Main Menu ---------
- A. General Condition
- B. Examine Body Area
- C. Auscultation
- D. Percussion

Select an option: {student presses the Esc key to return to the Main Tutoring Menu}

--------- Main Tutoring Menu ---------
- A. Patient record
- B. Symptoms
- C. Examinations
- D. Tests
- E. Hypothesis
- F. Management
- G. View Log
- H. Review case data separately
- I. Print current log

Select an option: **H (Hypothesis)**

Enter your new addition(s) to the current hypothesis:

**** glandular fever**

The problem which you have added to your differential diagnosis (listed below) is correct:

{This means that Glandular Fever is part of the tutor's differential diagnosis}

- Glandular fever
  (Incidence: 1.3 in 1000 patients visiting the surgery)

--» *Hypothesis Update*

{The user has correctly updated his hypothesis. The fact that Glandular Fever is part of the tutor's differential diagnosis, the student model has now been updated}

--------- Main Tutoring Menu ---------
A. Patient record
B. Symptoms
C. Examinations
D. Tests
E. Hypothesis
F. Management
G. View Log
H. Review case data separately
I. Print current log

Select an option: **M (Management)**

--» *Formulating a Management Plan*

{The management plan is going to be based on the two diseases in the user's differential diagnosis}

--------- Management Actions Menu ---------
Therapeutic Intervention
Diagnostic Intervention
Clinical Procedure
Refer to Other Agency
Review Later
Administrative Procedures

Select an option: **(Therapeutic Intervention)**

--------- Therapeutic Interventions Menu ---------
Drug Treatment
Physical Treatment
Non-Physical Treatment

Select an option: **(Drug Treatment)**

--------- BNF Categories ---------
Gastro-Intestinal Tract
Cardiovascular System
Respiratory System
Central Nervous System
Infections
Endocrine System
Obstetrics, Gynaecology & Urinary Tract
Malingnant Disease & Immunosuppression
Nutrition and Blood
Musculoskeletal System
Eye
Skin

Select an option: **(Central Nervous System)**

--------- Central Nervous System Menu ---------
Benzodiazepine Sedative
Benzodiazepine Anxiolytic
Hypnotic
Barbiturate
Antipsychotic
Lithium Salt
Antidepressant
Vestibular Sedative
Anti Nauseant
Simple Analgesic
Compound Analgesic
Narcotic Analgesic
Ergotamine
Anti Convulesant (non Barbiturate)
Anti Parkinsonian

Select an option: **(Simple Analgesic)**

Select an option: {student presses the Esc key to return to the BNF Categories Menu}

--------- BNF Categories ---------
Gastro-Intestinal Tract
Cardiovascular System
Respiratory System
Central Nervous System
Infections
Endocrine System
Obstetrics, Gynaecology & Urinary Tract
Malingnant Disease & Immunosuppression
Nutrition and Blood
Musculoskeletal System
Eye
Skin

Select an option: **(Infections)**

--------- Infections Menu ---------
Penicillin
Penicillin Esters
Cephalosporin
Tetracycline
Aminoglycoside
Erythromycin
Chloramphenicol
Sulphonamide
Co Trimoxazole
Trimethoprim
Anti Tuberculous Agents
Metronidazole
Antifungal Agents
Anti Viral Agents
Anti Malarial Agents
Anti Helminthics

Select an option: **(Penicillin)**

Select an option: {student presses the Esc key to return to the BNF Categories Menu}

--------- BNF Categories ---------
Gastro-Intestinal Tract
Cardiovascular System
Respiratory System
Central Nervous System
Infections
Endocrine System
Obstetrics, Gynaecology & Urinary Tract
Malingnant Disease & Immunosuppression
Nutrition and Blood
Musculoskeletal System
Eye
Skin

Select an option: {student presses the Esc key to return to the Therapeutic Intervestions Menu}

--------- Therapeutic Interventions Menu ---------
Drug Treatment
Physical Treatment
Non-Physical Treatment

Select an option: {student presses the Esc key to return to the Main Management Actions Menu}

--------- Management Actions Menu ---------
Therapeutic Intervention
Diagnostic Intervention
Clinical Procedure
Refer to Other Agency
Review Later
Administrative Procedures

Select an option: {student presses the Esc key to start processing the Management Plan}

> > Processing your management plan. Please wait ...

{The tutor starts to process the user's management plan. According to the madical database (Glandular Fever) the request for Blood test is not necessary! However, if the user requests the type of test, the tutor will report abnormal blood test results which should help the user to correctly diagnose the problem as Glandular Fever}

You are absolutely right in selecting the following mandatory action(s):

** Simple Analgesic
<u>Remarks</u>
All patients should receive symptomatic relief. Paracetamol for relief of fever and pain, with soluble aspirin gargles for a sore throat.

Your selection of the following action(s) is correct as a temporary treatment or management:

** Penicillin
Remarks
Acutely ill patients in whom a clinical differentiation between tonsillitis and Glandular Fever cannot be made may well be optimally treated with penicillin.

{The fact that the user selected the mandatory action(s) and no harmful or unnecessary ones, the full case management-section credits was added to the student model. The remarks shown after each management action are the comments entered by the expert when the medical database was built}

--» *Problem Diagnosis*

{Having managed the case, the user will now be questioned on the exact diagnosis of the problem}

Having managed the case, and on the basis of what you already have on this case, can you, at this stage, give the diagnosis for this problem (y/n) ? Y

Type in your diagnosis for this case:

** **viral pharyngitis**

No. The problem in this case was

Glandular fever

{The fact that the user did not specify the correct diagnosis for this problem, the student model has been decremented by the addition of the negative diagnosis credits}

> > Press any key to see how your selected problem and any other problem(s) in your hypothesis list were or can be eliminated ...

{The tutor displays the facts that the user requested and any others which he did not request that differentiates the main problem and the additional one the user considered in his hypothesis}

Note:
The values associated with each problem are the ones expected when such a problem exists. There may be some additional information in the following list which, if requested, should have confirmed the absence of the problem.

(*) Previously requested case data.

- Viral pharyngitis
** General Condition - Normal (*)
** Blood for Haematology: White Cell Count (Ref.Range 4500 - 11000)
** Blood for Haematology: Neutrophils (Ref.Range 712 - 7588)

The management plan for Glandular fever is as follows:

{The user is shown the managment plan of the main disease for reference purposes}

Harmful Actions

Penicillin Esters
Remarks:
NEVER give ampicillin or derivatives for sore throats as the complication of exfoliative dermatitis seen with these drugs in glandular fever can be lethal !!

Mandatory Actions

*Simple Analgesic
Remarks:
All patients should receive symptomatic relief. Paracetamol for relief of fever and pain, with soluble aspirin gargles for a sore throat.

Optional Actions

*Penicillin
Remarks:
Acutely ill patients in whom a clinical differentiation between tonsillitis and Glandular Fever cannot be made may well be optimally treated with penicillin.

General footnote: Any of the above actions marked with an '*' were part of your management plan.

In addition to the problems in your hypothesis, were there any other problems which you considered when the patient record and the presenting symptom(s) were displayed at the beginning of this session (y/n)? N

You're right. There were no other problems to be considered.

{The user has now completed the case. He is now offered a number of choices in the form of an End of Seesion Menu}

--------- End of Session Menu ---------
A. Print current case log
B. Display current case log
C. Save current log to permanent file
D. Review a particular problem
E. Cross reference problems by management actions
F. Cross reference problems by symptoms
G. Display student level graph
H. Print consultations report on present medical condition

Select an option: {student presses the Esc key to end the consultation}

»»»» End of Case ««««