Sabri A. Mahmoud and Mohammed M. Mandurah, “CATIB Environment and its applications in the development of Intelligent Computer Assisted Instruction(ICAI).” Proceedings of the Second Symposium of the Arabization of Computers at King Saud University, Riyadh, 27-30 March, 1994, Vol. 1, pp. 47-94. (Invited paper in Arabic.)

Keywords: CATIB, Arabic Programming Languages, Course Ware Development, Intelligent Computer Assisted Instruction.

بيئة البرمجة "كاتب" وتطبيقاتها في تطوير نظم تعليم حاسوبية ذكية*

صبري عبد الله محمود و محمد محمود مندورة

كلية علوم الحاسب والمعلومات

جامعة الملك سعود

ص.ب ٥١١٧٨ - الرياض ١١٥٤٣

الملخص :

تناقش هذه الدراسة تأثير التطورات في تقنيات الحاسوب والبرمجيات على نظم التعليم بمساعدة الحاسوب . فالتطورات في أساليب الذكاء الإصطناعي ونظم الخبرة وقواعد المعرفة والنصوص الفوقية وتقنيات التخاطب بين الإنسان والآلة يُتوقع لها أن توجد الكثير من الحلول للمشكلات التي كانت تعاني منها النظم الحاسوبية التعليمية التقليدية . وعلى الأخص يُتوقّع أن تؤثر هذه التقنيات بشكل مباشر على مكونات النظم التعليمية وهي : سيناريوهات التعلّم ، تمثيل مجال المعرفة ، نمذجة الطلاب ، تشخيص الخلل المعرفي لدى الطلاب ، توليد المسائل ، وإدارة التفرعات في البرنامج التعليمي . وتستعرض الدراسة بيئة البرمجة العربية "كاتب" المصمّمة لغرض إعداد الدروس التعليمية ، وخصائصها المتميزة التي تيسر تطوير تطبيقات ذكية باستخدام أساليب الذكاء الإصطناعي ، وكذلك تعطي أمثلة على بناء نظم خبيرة باستخدام أوامر "كاتب" .

---

* هذا المشروع مدعوم من مدينة الملك عبد العزيز للعلوم والتقنية - مشروع أت- ١١- ٤٢

١- مقدمة :

بدأت التجارب والمحاولات لتطوير استخدامات الحاسوب التعليمية منذ مدة طوية في بداية الستينات من هذا القرن . ولكن واقع تقنيات الحاسوب من أجهزة ولغات برمجة في تلك الحقبة من الزمن لم يكن مواتياً لإنتاج نظم حاسوبية تعليمية فعّالة يمكن استخدامها على نطاق واسع . فأجهزة الحاسوب كانت مكلفة جداً ، بحيث إقتصرت الاستفادة من خدمات الحاسوب على المنشآت الكبيرة القادرة على تحمل تكاليف الحاسوب . وكانت لغات البرمجة صعبة الاستخدام بحيث لم يكن يقدر على البرمجة بها إلا المبرمجون المتخصصون . كذلك كانت تلك اللغات مصممة لإجراء العمليات الحسابية ، أما قدرتها على معالجة النصوص فكانت محدودة جداً .

ولكن تلك الصعوبات لم تثن الباحثين عن محاولة تحقيق طموحاتهم في تطوير النظم التعليمية بمساندة الحاسوب . وكان إتجاه التطوير في محاولة تصميم أجهزة وبرمجيات خاصة لغرض الاستخدامات التعليمية . فظهر في السبعينات نظام (Plato) الذي يتضمن شاشات خاصة حساسة يمكنها أن تستجيب للمس بواسطة الأصابع كوسيلة لإدخال الأوامر والمعلومات . كذلك ظهر نظام (IBM-1500) المزود بتجهيزات خاصة تمكنه من عرض الأفلام وتشغيل أشرطة التسجيل الصوتية [٢.١] .

أما في مجال لغات البرمجة وبرمجيات النظم ، فكان التوجه نحو تطوير لغات خاصة للتطبيقات التعليمية وكذلك نظم خاصة لإعداد الدروس . وكانت جميع تلك الجهود تهدف إلى التغلب على صعوبة استخدام لغات البرمجة السائدة في تلك الحقبة ، والتي كانت لاتخدم كثيراً متطلبات البرمجيات التعليمية . فظهرت لغات البرمجة التعليمية الخاصة مثل (Pilot) و (Course Writer) و (Tutor) ، ونظم إعداد الدروس مثل (VAULT) و (EAASY) و (AIS) و (CourseMaker) [٣-٧] .

ولكن تقنيات الحاسوب ونظم التشغيل ولغات البرمجة في الستينات والسبعينات ومطلع الثمانينات لم تكن في مستوى يسمح بتطوير برمجيات تعليمية ذات قدرات تخاطبية عالية ، أو ذات قدرات تسمح لها بالتعامل مع مختلف مستويات الطلاب . والأسلوب السائد في تطوير معظم البرمجيات التعليمية في تلك الفترة كان يتبع منهج التعليم المبرمج (programmed Instruction) . وهذا هو الأسلوب المعروف الذي يتبع النموذج (إشرح - إسأل - قوّم - تفرّع حسب نتيجة التقويم) . كذلك كانت معظم البرمجيات التعليمية عبارة عن شاشات متتالية من النصوص أو الأسئلة دون أن تحتوي على أنماط إبداعية تجديدية .

بمعنى آخر ، كانت معظم البرمجيات التعليمية مجرد «تقليب صفحات كتاب إلكتروني» .

ولكن السنوات الأخيرة شهدت تطوراً كبيراً في علوم الحاسوب وتقنياته وبرمجياته . وعلى الأخص يعتبر تطوير نظام التشغيل (Windows) وانتشار استخدامه بين المستخدمين منعطفاً بارزاً لما يتمتع به من خصائص تيسر عمليات التخاطب بين الإنسان والآلة . كذلك نضجت الكثير من مفاهيم وأساليب الذكاء الإصطناعي مما أخرجها من حيز التجريب إلى حيز التطبيق . وعلى الأخص يمكن ذكر التطور في مجال قواعد المعرفة والنظم الخبيرة . وأيضاً من أهم ملامح التطور في السنوات الأخيرة هو تبلور أسلوب النصوص الفرقية (hypertexts) كأسلوب لحفظ المعرفة وتمثيلها . وهذا الأسلوب ينظم عمليات حفظ المعلومات والربط بينها مما ييسّر عمليات استرجاعها .

في هذا البحث سنناقش تأثير التطور في تقنيات وأساليب الذكاء الإصطناعي على نظم التعليم بمساعدة الحاسوب ، وكذلك تأثير التطور في أساليب النصوص الفرقية على تمثيل المعرفة وتصميم قواعد المعرفة . وفي الجزء الأخير من البحث سنستعرض التطور في لغة البرمجة العربية «كاتب» ، وسنعطي أمثلة على كيفية بناء نظم تعليمية ذكية باستخدام الإيعازات الخاصة في اللغة .

١ - الذكاء الإصطناعي ونظم التعليم بمساعدة الحاسوب :

إن أحد أبرز معالم التطورفي علوم وتقنيات الحاسوب في السنوات الأخيرة هو نضوج أساليب الذكاء الإصطناعي وخروجها من حيز التجريب إلى حيز تطوير التطبيقات المفيدة . وإن أحد المجالات التي يتوقع أن تجني ثمار ذلك التطور هو مجال استخدامات الحاسوب في مساندة التعليم [٧-١٨] .

ويمكن لتقنيات وأساليب الذكاء الإصطناعي أن تؤثر مباشرة على مكونات أي نظام حاسوبي لمساندة التعليم وهي [١٢] :

- سيناريوهات التعلّم (Learning Scenarios) .
- تمثيل معرفة المجال (Domain Knowledge Representation) .
- نمذجة الطالب (Student Modeling) .
- تشخيص الخلل المعرفي لدى الطالب (Student Diagnosis) .
- المعرفة المتعلقة بأصول التدريس (Pedagogical Knowledge) .

- إدارة التفرعات (Discourse Management) .
- توليد المسائل (Problem Generation) .
- بيئة التخاطب مع المستخدم (User Interface) .

وسنناقش فيما يلي تأثير الذكاء الإصطناعي على المكونات المذكورة أعلاه .

٢-١ سيناريوهات التعلّم (Learning Scenarios) :

إن المحور الأساسي الذي تتغير وفقه سيناريوهات التعلم هو توزيع الأدوار بين الطالب والمدرس في العملية التعليمية . وعلى طرف من هذا المحور ، يمكن أن تتبع النظم الحاسوبية التعليمية أسلوب (شرح-مثال-سؤال) ، وتكون الدروس التعليمية مجرد إطارات ثابتة من النصوص التعليمية . وفي هذا السيناريو يكون دور الطالب في التحكم في العملية التعليمية محدوداً ، ولايستطيع الطالب أن يحيد عن تعلّم مايُعرض له أو أن يحاول استكشاف أشياء جديدة . وفي مقابل ذلك على الطرف الآخر من المحور ، يمكن أن يكون سيناريو التعلّم من نوع التعلّم الاستكشافي مثل التعلّم من خلال البرمجة بلغة «لوجو» ، أو النمط الذي يتيح للطالب أن يختار المواضيع والمادة العلمية التي يرغب في تعلمها [١٣، ١٤] .

ويكمن دور أساليب الذكاء الإصطناعي في سيناريوهات التعلّم في تحقيق التوازن المثالي عند توزيع الأدوار بين الطالب والمدرس في العملية التعليمية . فالمدرّس الذكي الناجح يجيد توزيع هذه الأدوار ، ويعرف متى يطلب من الطلاب تعلّم مواضيع محدّدة ، ومتى يتيح لهم فرصة التعلّم الاستكشافي . وكذلك ينبغي أن تكون نظم التعليم الحاسوبية الذكية .

٢-٢ تمثيل معرفة المجال (Domain Knowledge Representation) :

كانت أحد أبرز جوانب الضعف في النظم الحاسوبية التعليمية التقليدية هو القيود الشديدة على تمثيل المعرفة فيها . فالمعرفة «معلّبة» في الإطارات التي تعرضها بطريقة يصعب فيها عرضها بأساليب أخرى ، أو الاستفادة من هذه المعرفة بطرق أخرى . كذلك الحال بالنسبة للمسائل والتمارين والأجوبة عليها ، وبالنسبة للتفرعات في البرنامج ، فجميعها يأتي «معلّباً» لاسبيل لتطويره أو تعديله أو الاستفادة منه في غير الإطار الذي صُمم لأجله . لذا كان أحد أهم أهداف النظم الحاسوبية التعليمية الذكية هو تطوير أساليب تمثيل المعرفة بما يمكّن من استغلالها كمعرفة حقاً . بمعنى آخر أن هذه المعرفة يجب أن تنعكس على سير العملية

التعليمية وعلى جميع جوانبها [١٥] .

وتُبذل جهود بحثية كثيفة للبحث عن أفضل السبل لتمثيل المعرفة للأغراض التعليمية . وقد تتغير طريقة تمثيل المعرفة بحسب المجال المعرفي ، فالمعرفة نفسها يجب أن تحدد لنا أنجع السبل لتمثيلها . ويجب أن يستفاد من المعرفة ذاتها في توليد الأسئلة والتعامل مع إجابات الطالب على الأسئلة والإمتحانات ، وتحليل هذه الأجوبة بهدف معرفة أماكن الخلل المعرفي لدى الطالب ، ثم التعامل مع أي وضع جديد بما يناسب من عرض مادة معرفية أو التفرع إلى نقطة مناسبة في العملية التعليمية [١٢] .

٢-٣ نمذجة الطالب (Student Modeling) :

يمكن تعريف نموذج الطالب بأنه تحديد ووصف لقدرات الطالب ومعرفته بحيث يمكن للبرنامج الحاسوبي التعليمي أن يستفيد من هذا التحديد والوصف في تحقيق الهدف التعليمي بفعالية . ففي الوضع المثالي ينبغي على المدرس الناجح أن يعرف المستوى المعرفي لكل طالب ، ويعرف قدرات كل واحد منهم ، وماهي أنسب الطرق لإيصال المعلومات لهم ، بل ويعرف ماهي التطلعات والطموحات المستقبلية لكل طالب بحيث يوجه مسار تعلّمه بما يناسبه [١٥] .

وماذكر أعلاه يشكل تحدياً من التحديات التي تواجه استخدامات الذكاء الإصطناعي في نظم التعليم بمساندة الحاسوب . والمشكلة ذات شقين : الأول يتعلق بكيفية تطوير نموذج الطالب ، والثانية تتعلق بكيفية الاستفادة من نموذج الطالب في الإرتقاء بالعملية التعليمية .

٢-٤ تشخيص الخلل المعرفي لدى الطالب (Student Diagnosis) :

تشبه عملية تشخيص الخلل المعرفي لدى الطالب أي عملية تشخيص أخرى مثل : تشخيص الأمراض أو تشخيص الأعطال في الأجهزة . فالأخطاء التي يرتكبها الطالب دلالة على وجود قصور أو خلل معرفي في مكان ما لدى الطالب . والهدف من عملية التشخيص هو تحليل إجابات الطالب وأخطائه للوصول إلى مسببات تلك الأخطاء [٩] . وترتبط عملية تشخيص الخلل المعرفي لدى الطالب إرتباطاً وثيقاً بعملية نمذجة الطالب . فنتائج عملية التشخيص ستكون عناصر في نموذج الطالب . وفي الواقع فإن جميع مكونات العلمية التعليمية ترتبط مع بعضها البعض . فالمعرفة في مجال ما تحدد مانريد للطالب أن يتعلمه ، وهذه بدورها هي المعيار الذي نقيس به مستوى أداء الطالب ، وتحليل أداء الطالب وأخطائه يقودنا لمعرفة أماكن الخلل

المعرفي لدى الطالب ، وهكذا .

٢-٥ المعرفة المتعلقة بأصول التدريس (Pedagogical Knowledge) :

والذي تحتاجه نظم التعليم بمساعدة الحاسوب من هذه المعرفة هو الاستراتيجيات التعليمية المتعلقة بإختيار أفضل سبل عرض المعلومات ، تحقيق التوازن في تحكم المدرس في العملية ، تحديد كمية التغذية الراجعة ، وتحديد أساليب تقويم أداء الطالب [١٤] . فالمعرفة بأصول التدريس هي التي تحدد هذه السياسات. وهنا يظهر بوضوح الإرتباط بين مكونات العملية التعليمية في النظم الذكية . فهذه المعرفة هي الروح لجميع العمليات الأخرى ، وقد تكون مخزنة في قاعدة معرفة محددة في النظام ، أو قد تكون موزعة، وفي كلتا الحالتين فإن هذه المعرفة هي التي توجه القرارات في العمليات الأخرى .

٢-٦ إدارة التفرعات (Discourse Management) :

إن المدرس الجيد يسمح لمسار العملية التعليمية أن يتوجه وفق إحتياجات الطالب ومتطلباته المعرفية وليس وفق مسار محدد مبرمج . وإدارة التفرعات في النظم الحاسوبية التعليمية الذكية هي المسؤولة عن تحقيق هذا الهدف ، وهذا يتطلب أن تكون إدارة التفرعات مرنة وقادرة على التجاوب مع مختلف الظروف والحالات . وأحد طرق تصميم إدارة التفرعات أن تكون حول نموذج للطالب وحول شبكة من المتطلبات المعرفية السابقة المبنية على هيئة شجرة متكاملة . فإعتماداً على نموذج الطالب يحدد النظام ما إذا كان الطالب يمكنه الإنتقال إلى الموضوع التالي أو تقوية فهمه في الموضوع الحالي [٩] . وشجرة المتطلبات السابقة هي التي تحدد ماهي المواضيع التالية التي ينبغي عرضها للطالب .

٢-٧ توليد المسائل (Problem Generation) :

إن أحد جوانب القصور في النظم التقليدية للتعليم بمساعدة الحاسوب أن المسائل والأجوبة تكون «معلبة» - أي جاهزة ولاتتولد تلقائياً بناء على الحالة أو الظرف أو إحتياج الطالب . ثم تطورت عملية توليد المسائل لتكتسب بعض القدرة على توليد مسائل مختلفة بحيث يكون التغير في عناصر محددة في المسائل . وتعتمد هذه الطريقة على تخزين صيغ للمسائل بحيث تكون عناصرها على هيئة متغيرات قابلة لإكتساب قيم مختلفة . وعند توليد المسائل يتم استبدال القيم المختلفة بالمتغيرات [٨] .

أما غاية الذكاء الإصطناعي فهي توليد المسائل إعتماداً على فهم أعمق للمعرفة في المجال . ويعتمد ذلك على الطريقة المتبعة في تمثيل المعرفة ، ثم كيفية الاستفادة من ذلك في توليد المسائل .

٢-٨ بيئة التخاطب مع المستخدم (User Interface) :

أظهرت إحدى الدراسات أن الإنسان يتذكر ٢٥٪ مما يسمعه ، و٤٥٪ مما يراه ويسمعه ، و٧٠٪ مما يراه ويسمعه وينفذه أيضاً [١٦] . كذلك يعرف بعضنا العبارة الشائعة «أن الصورة تعدل ألف كلمة» . وهذه جميعاً تؤكد على أهمية أن تتضمن البرمجيات التعليمية مزيجاً من النصوص والصور والرسوم والأصوات لكي تحقق أهدافها بكفاءة أعلى .

أما تطبيقات الذكاء الإصطناعي في تطوير بيئة التخاطب بين الإنسان والآلة فتكمن في تطوير تطبيقات المعالجة الآلية للغات الطبيعية ، ويشمل ذلك فهم النص المكتوب كوسيلة آلية لإدخال المعلومات للحاسوب، وأيضاً توليد النصوص كوسيلة لإخراج المعلومات وعرضها [٩ ،١٧ ،١٨] . وهذه المجالات لاتزال ميداناً خصباً للجهود البحثية والتطويرية . ويندرج تحت هذه الجهود محاولات فهم الكلام أو توليد الكلام (Speech Recognition and Speech Synthesis) حيث الهدف منها أن يتخاطب الإنسان مع الآلة بالكلام الطبيعي .

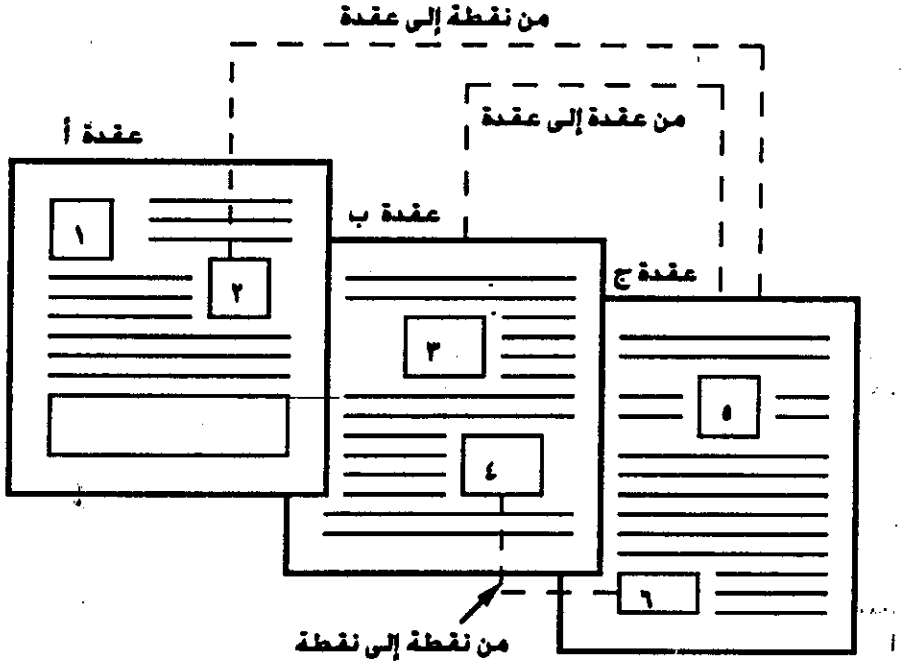
٣ - النصوص الفوقية وتمثيل المعرفة :

من المعروف أن بعض المطبوعات كمقالات المجلات وادلة المستخدمين الفنية تمتاز بترابط أجزائها الملموسة والمنطقية .فمن الناحية الملموسة يتكون المقال من سلسلة طويلة من الكلمات الموزعة على الأسطر والصفحات . ومن الناحية المنطقية فهي كلمات جمعت لتكوّن جُمل ، وتتجمع الجمل لتكوّن فقرات ، والفقرات تكوّن الفصول ...الخ .

الا أنه في بعض المطبوعات كالقواميس والمعاجم يكون ترابط الاجزاء الملموسة منفصلاً عن ترابط الاجزاء المنطقية . فمن حيث التركيب فانها تتكون من وحدات متسلسلة كتسلسل الكلمات والفقرات في مقالة أو كتاب ، أما من الناحية المنطقية فهي أكثر تعقيدا . فالقارئ لا يقرأ هذه المطبوعات من البداية للنهاية ، ولكنه يبحث فيها عن مقال معين أو معلومة معينة بشكل عشوائي ، ثم يقرأ الجزء الذي تم

اختياره بالتسلسل . ثم يجد القارئ أن عليه الانتقال لكلمات أخرى ذات علاقة ، وحتى يتابع القارئ عليه أن يتبع المؤشرات الى الوثيقة المطلوبة ثم الى الموضوع المطلوب ثم الى الجزء المطلوب . وهذه الوثائق لها تسلسل يسهل عملية البحث عنها ، ولها مسارات منطقية مرتبطة بشبكة تربط مجموعة من الوثائق او المواضيع ... الخ . وهذا الأسلوب في حفظ المعلومات وربطها ببعضها البعض كان هو المنطلق وراء تطوير ما يعرف بالنصوص الفوقية (hypertexts) .

ففي النصوص الفوقية يتم خزن المعلومات في عقد (nodes) مرتبطة فيما بينها بوصلات (links). وتمثل كل عقدة جزءا صغيرا من معجم أو ربما صورة مع شرح بسيط عنها أو ملفاً كاملا أو وحدات معلومات في قاعدة معلومات [١٩-٢١] . ويوضح الشكل (١) مثالا لوثيقة ذات نصوص فوقية . وتمثل الوصلات مؤشرات الى أماكن العقد . ويمكن أن تحتوي عقد النصوص الفوقية على رسائل صوتية أو صور أو مقاطع من أفلام سينمائية وربما برامج حاسوب يمكن تنفيذها عند اختيار العقدة التي فيها البرنامج . وكما يتضح فإن النصوص الفوقية تمثل أسلوباً مرناً في تمثيل المعرفة وحفظها . وهذا الأسلوب يتناسب مع التطبيقات التعليمية - خاصة مع الأنماط المصممة وفق مبدأ التعلّم الاستكشافي الذي يتم بدرجة كبيرة تحت تحكم المستخدم نفسه .



شكل (١) : مثال لوثيقة تحتوي على نصوص فوقية .

٤ - بيئة البرمجة بلغة "كاتب" :

تعتبر لغة "كاتب" احدى لغات البرمجة العربية التي تم تصميمها لتيسير تحضير الدروس باستخدام الحاسوب ، وتحتوي على ميزات عديدة موجهة لخدمة ذلك الهدف [٢٢] . وتسهّل القواعد اللغوية البسيطة للغة "كاتب" تعلمها من غير المبرمجين وكذلك استخدامها في تحضير الدروس . وقد بدأ العمل في تطوير اللغة سنة ١٩٨٤م وتم تصميم وتطوير النسخة الاولى منها للعمل على نظام تشغيل (CPM) الذي كان مشهورا ذلك الوقت [٢٣] . وقد تم الحصول علي منحة من مدينة الملك عبدالعزيز للعلوم والتقنية سنة ١٩٩١م لنقل اللغة الى الاجهزة المعتمدة على نظام تشغيل دوس (DOS) ودعمها بامكانات جديدة واضافة محررنصوص لها يستفيد من التطور في تقنية الحواسيب [٢٤] .

وقدتم تصميم النسخة الاولى من "كاتب" على غرار لغة بايلوت (Pilot)[٦] . و"كاتب" لغة مترجمة (interpreted) وليست لغة مجمعة (compiled) حيث يتم تنفيذ كل سطر عند قراءته . ويتكون برنامج "كاتب" من أوامر وإيعازات بحيث يكون كل أمر أو إيعاز على سطر منفصل . وصيغة أسطر أوامر "كاتب" كما يلي :

[أمر] [ن/ل] : [رسالة | تعبير]

[أمر] هو أحد أوامر "كاتب" وهو اختياري إذ ليس من الضروري أن يحتوي كل سطر على أمر من أوامر "كاتب" . وفي حالة عدم وجود الأمر ، يقوم المترجم بتكرار الأمر السابق .

[ن/ل] نعم/لا. حيث يعتمد تنفيذ الامر في هذا السطر على نتيجة مقارنة سابقة .

رسالة : هي مجموعة من الكلمات وقد تحتوي على أسماء متغيرات .

تعبير : هو تعبير من تعابير "كاتب" الرياضية أو المنطقية .

ويظهر الشكل (٢) مثالاً لبرنامج مكتوب بلغة "كاتب" .

وبالنظر للمثال نجد أول أمر وهو "ملاحظة" يستخدم لكتابة ملاحظات المبرمج ، وهي تهمل من قبل

مترجم "كاتب . يلي ذلك أمر "إمسح-شاشة" حيث يقوم المترجم بمسح محتويات الشاشة . يلي ذلك في السطر التالي أمر "اكتب وسط" يقوم المترجم بعرض الرسالة "بسم الله الرحمن الرحيم" في وسط السطر . أما السطر الثاني فإنه لا يحتوي على أي أمر ، لذلك يقوم المترجم بتكرار الامر السابق أي "اكتب-وسط" . ثم يلي ذلك ثلاثة أسطر لكتابة رسائل على الشاشة . ويلي ذلك أمر "نهاية" حيث يقوم المترجم بانهاء البرنامج والعودة الى بيئة "كاتب .

```
ملاحظة      : مثال بسيط لبرنامج كاتب
امسح شاشة:
اكتب-وسط : بسم الله الرحمن الرحيم
          : مثال ١
اكتب      : تمتاز لغة كاتب ببساطتها وسهولة استخدامها .
          : وتتضمن محرر للنصوص متكامل معها سهل الاستخدام
          : وله إمكانات متطورة .
نهاية     :
```

شكل (٢) : مثال لبرنامج مكتوب بلغة "كاتب .

وتحتوي النسخة المعدلة من لغة "كاتب على عدة تحسينات . وقد تم الاحتفاظ بالصيغة المبسطة لقواعد اللغة كما في النسخة الاصلية ، ولكن تم اضافة اوامر كثيرة جديدة تسهل على المبرمج إعداد دروس باستخدام وسائل متطورة كالنصوص الفرقية والذكاء الصناعي والرسم .. الخ . ويوضح الملحق (أ) قائمة بأوامر "كاتب [٢٥] . وقد تم تقسيمها الى ثمانية أقسام وهي:

(١) أوامر الادخال والاخراج .

(٢) أوامر التحكم بالمسار .

(٣) أوامر التحكم بالشاشة .

(٤) أوامر معالجة حزم الحروف .

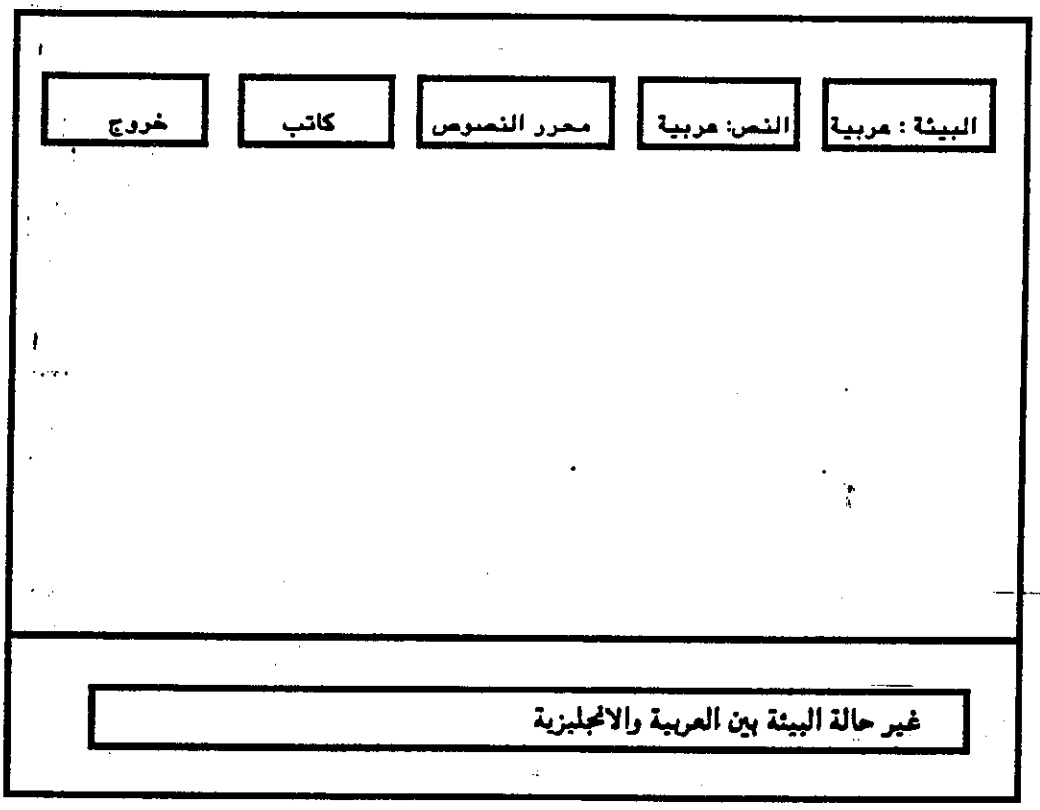
(٥) الأوامر الرياضية والمنطقية .

(٦) أوامر الرسم .

(٧) أوامر الملفات .

(٨) أوامر التحكم في بيئة "كاتب .

وتتضمن النسخة المطورة من "كاتب محرراً للنصوص بحيث يشكلان بيئة متكاملة للبرمجة . وفي داخل البيئة يمكن بسهولة الانتقال من المحرر الى اللغة أو تنفيذ برنامج مكتوب بلغة "كاتب من داخل المحرر . وعند تنفيذ أحد البرامج ، ووجد المترجم خطأ في أحد أسطر البرنامج ، فإنه يستدعي المحرر ويحمّل البرنامج ثم ينقل المشيرة الى مكان الخطأ مما يسهل على المستخدم تصحيح الاخطاء بسهولة ويسر .



شكل (٣) : بيئة العمل للغة "كاتب

بسطة

دروس

بأوامر

ويبين الشكل (٣) رسماً توضيحياً لشاشة الإفتتاح بعد تحميل "كاتب . وتظهر الشاشة الخدمات المتوفرة في بيئة البرمجة ، حيث يمكن للمستخدم إختيار لغة التخاطب عربي/إنجليزي ، إتجاه الكتابة على الشاشة (من اليمين إلى اليسار أو بالعكس) ، طلب محرر النصوص ، أو طلب "كاتب .

٥ – أوامر "كاتب الخاصة :

سنعرض فيما يلي بعض أوامر "كاتب التي طورت لتسهيل عمل الدروس . وهذه الاوامر غير متوفرة في معظم لغات البرمجة الاخرى :

١. اكتب-وسط : [رسالة]
اطبع-وسط : [رسالة]

حيث رسالة تحتوي على مجموعة من الكلمات وقد تحتوي على أسماء متغيرات صحيحة أو حقيقية أو حرفية .

يقوم مترجم "كاتب بعرض الرسالة على الشاشة (او طباعتها على الطابعة) في وسط السطر الحالي ، ويقوم باستبدال اسماء المتغيرات بقيمها . وهذا مما يسهل على المستخدم وضع العناوين الرئيسية أو الكتابة في الوسط .

٢. تذييل : [رسالة]

يقوم مترجم "كاتب عند تنفيذ هذا الامر بالانتقال الى أسفل الشاشة وعرض "رسالة" ثم ينتظر أن يضغط المستخدم زر "ادخل" حتى ينتقل لتنفيذ للامر التالي . وقد تحتوي "رسالة" على أسماء متغيرات يتم استبدالها عن عرضها بقيم المتغيرات كما هو الحال في أمر "اكتب-وسط" . وفي حالة عدم وجود "رسالة" يقوم المترجم بعرض رسالة "اضغط مفتاح أدخل للاستمرار" ثم ينتظر حتى يضغط المستخدم زر "ادخل" .

٣. صفحة :

يقوم مترجم "كاتب بعرض اختيارين في أسفل الشاشة ، احدهما للاستمرار والآخر العودة للصفحة السابقة . وحسب اختيار المستخدم يتم الاستمرار في الصفحة التالية أو العودة

للصفحة السابقة .

٤. دحرج : [[@]عدد]

يقوم أمر دحرج بدحرجة الشاشة عدداً من الأسطر مقدارها "عدد" . و"عدد" اما متغير صحيح أو عدد صحيح . واذا لم يذكر أي رقم أو اسم متغير بعد النقطتين يتم دحرجة الشاشة سطرا واحدا.

٥. قائمة : [[$]رسالة] ، [[$]اسم-قائمة] ، @ اسم متغير ،[$]اختيار ١ ،[$]اختيارات .....

قائمة : [[$]رسالة] ، [[$]اسم-قائمة] ، $ اسم-متغير

وتحتوي رسالة على حزمة من الحروف أو اسم أو اكثر من اسماء المتغيرات . وجميع الحروف تظهر كما هي ما عدا أسماء المتغيرات فتستبدل بقيمها .

اسم-قائمة : اسم اختياري للقائمة

اسم-متغير : هو اسم متغير يتم تخزين الاجابة المدخلة به ،

اختيار١ ، اختيار٢ ... : هي بدائل القائمة وهي مجموعة من الاختيارات (الاستجابات) المحتملة لرسالة التوجيه . وتظهر على شكل قائمة . واذا لم يتم ذكر أية أختيارات تظهر الإشارة «=>» ويقوم المستخدم بادخال الجواب المطلوب .

مثال :

قائمة : ادخل الاختيار المطلوب ، القائمة الرئيسية ، $ جواب ، أ- تغيير لغة ، ب- كاتب ، ج- محرر ، د- نهاية .

عند تنفيذ الامر ، تظهر الشاشة كما يلي :

القائمة الرئيسية
أ – تغيير لغة
ب– كاتب
ج–محرر
د– نهاية
ادخل الاختيار المطلوب [ ]

٦. طابق : نموذج ١ [|نموذج ٢| ... نموذج ن]

طابق-اقفز : نموذج ١ [|نموذج ٢| ...| نموذج ن]

نموذج ١ ... ن : كلمة أو مجموعة كلمات

يقوم الامر طابق بمقارنة آخر إستجابة في أمر "خذ" والمحفوظة في الذاكرة الخاصة بأمر "خذ" مع النماذج المذكورة . فإذا وجد تطابق مع أحد هذه النماذج يقوم مترجم "كاتب برفع رايه "نعم/لا" وإلا فإنه سيتم خفقها . إن هذه النماذج هي ثابت حزمه .

تستعمل علامة "|" للفصل بين النماذج .

وأمر "طابق-اقفز" مشابه لامر "طابق" ، إلا أنه اذا لم يوجد تطابق مع أحد هذه النماذج يتم القفز إلى أمر "طابق" أو "طابق-اقفز" التالي . وجود أكثر من فراغ في النماذج بين الكلمات يتم اعتباره فراغا واحدا .

٧. منهاج-عشوائي : [*]اسم-منهاج١ | [*]اسم-منهاج٢|... | [*]اسم-منهاج ن

اسم -منهاج هو ثابت حزمه والنجمة إختيارية .

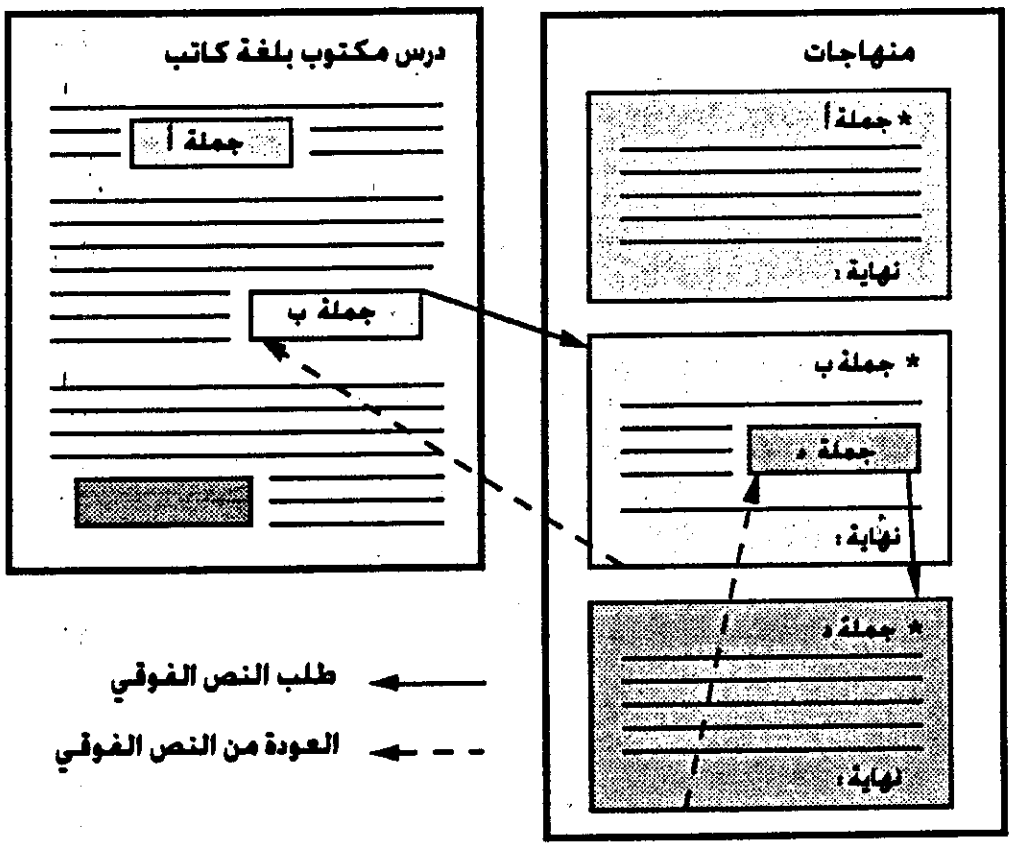
يؤدي هذا الامر الى استدعاء عشوائي لاحد هذه المنهاجات المحددة بعد ":" ، حيث يتم أولا تحديد عدد المنهاجات المذكورة في الامر ثم توليد رقم عشوائي يكون ضمن هذا العدد من المنهاجات . يقوم مترجم "كاتب بالانتقال الى المنهاج الذي يقابل الرقم العشوائي بعد حفظ رقم السطر التالي لهذا الامر .

٨. قل : [$]اسم ملف

اسم ملف هو اسم الملف المراد فتحه وقراءة ما فيه وتحويله الى نظام الاخراج الصوتي ويمكن التعبير عنه باسم متغير حزمة إذا كان يبدأ ب "$" أو ثابت حزمة . يتم ارسال الرسالة الصوتية المحفوظة في الملف المعطى الى نظام الاخراج الصوتي . وسيتم شرح الاوامر "عمق" و "أمامي" و"رجوعي" عند الحديث عن امكانات "كاتب من حيث استخدام النصوص الفوقية والذكاء الاصطناعي .

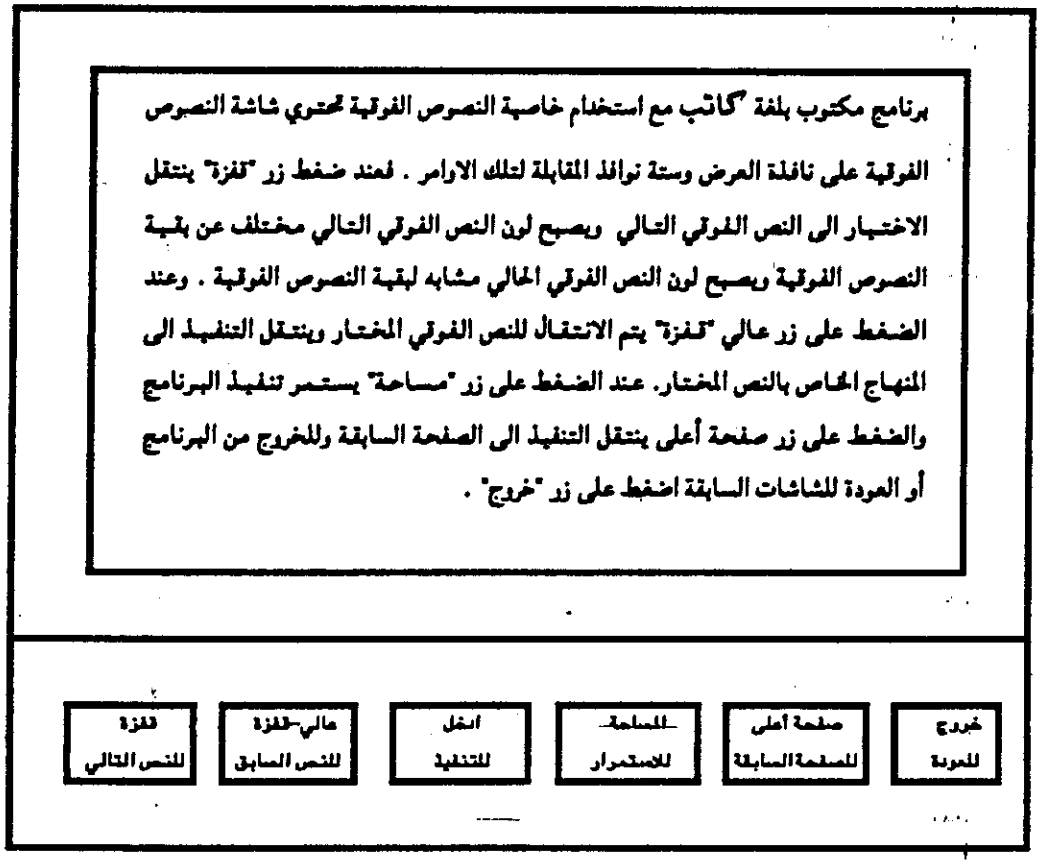
٦ - معالجة النصوص الفوقية في كاتب :

روعي في تصميم لغة كاتب أن تتم اضافة تحسينات خاصة منها اضافة النصوص الفوقية . وقد روعي أن تتم الاضافات بحيث تسهل البرمجة على المستخدم ، ولذا فقد تم اعتبار أي نص من النصوص الفوقية كعنوان ، وتمت معاملة هذه العناوين كما يعامل عنوان المنهاج (البريج) والتي اذا نفذت،يتم استدعاء البرنامج وتنفيذه . فحينما يختار المستخدم كلمة أو جملة من النصوص الفوقية فإن مترجم كاتب يبحث عن رقم السطر الذي يحتوي على العنوان المماثل لكلمة (أو جملة) النص الفوقي . فإن وجد النص الفوقي المطلوب ، فإن مترجم كاتب يقوم بتنفيذ كل الاوامر التالية لذلك العنوان . وقبل أن يتم التنفيذ للعنوان المطلوب يحتفظ كاتب برقم السطر الذي يلي السطر الحالي حتى يمكن الرجوع اليه .



شكل (٤) : تركيب الجُمَل والنصوص الفوقية في كاتب

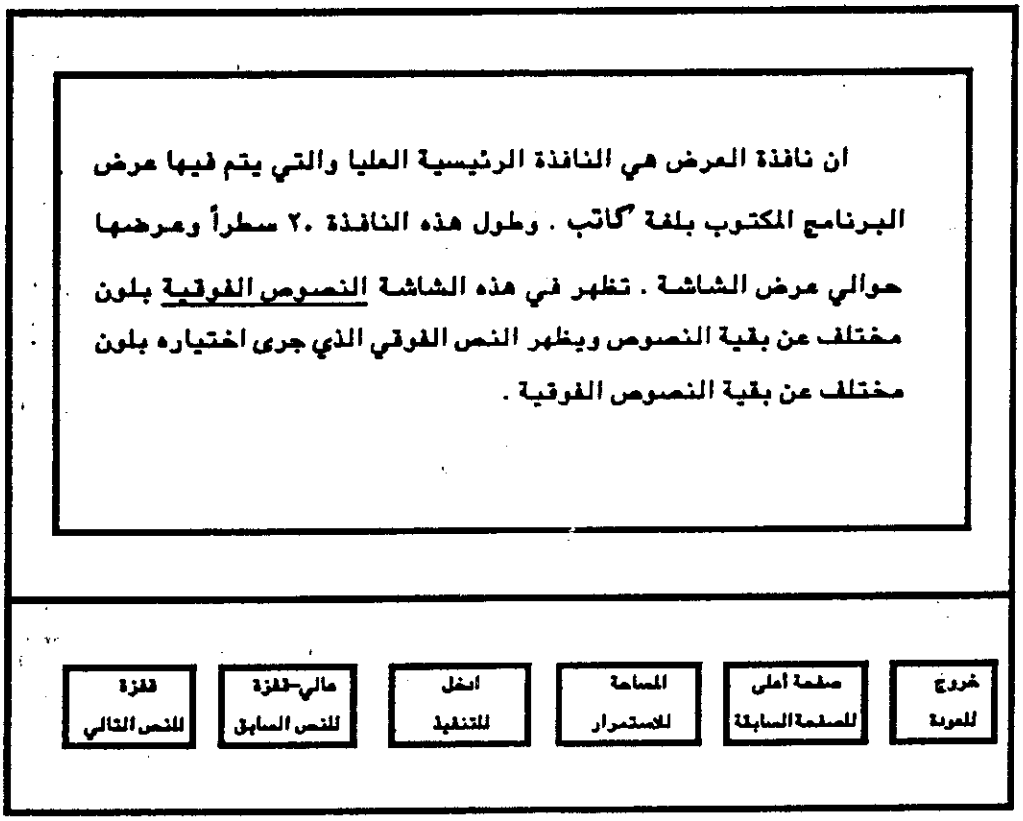
وهكذا يتم تنفيذ الاوامر التالية للعنوان وحتى الوصول الى أمر "نهاية" ، وعندها يقوم المترجم بالعودة الى السطر الذي تم حفظه (أي السطر التالي للسطر الذي تم استدعاء النص الفوقي فيه) . ويبين الشكل (٤) التركيب الذي تم على أساسه تنفيذ النصوص الفوقية في لغة "كاتب . كما يظهرالشكلان (٥) و(٦) مثالا لتطبيق النصوص الفوقية في لغة "كاتب . ففي الشكل (٥) يظهر البرنامج الرئيسي لهذا المثال والذي تظهر فيه كلمات النصوص الفوقية بلون مختلف عن بقية النص على الشاشة . وتميز في الشكل جُمَل النصوص الفوقية بخط تحتي . كما أن النص الفوقي الذي يشير اليه المؤشر حاليا (وهو النص المختار) يكون بلون مختلف عن بقية النصوص الفوقية على الشاشة ويُميّز في الشكل بخط تحتي سميك.



شكل (٥) : مثال لنص يحتوي على جُمَل ذات نصوص فوقية .

ويظهر الشكل (٥) أن النص الفوقي الذي تم اختياره هو "نافذة العرض" . وحينما يضغط

المستخدم زر "أدخل" يقوم مترجم "كاتب بالاحتفاظ برقم أول سطر في الصفحة الحالية ثم أخذ النص الفوقي "نافذة العرض" والبحث في البرنامج عن عنوان مشابه لهذا النص ، فإن وجده انتقل تنفيذ البرنامج الى المنهاج ذي العنوان المطلوب . ويظهر تنفيذ هذا النص الفوقي في الشكل (٦). ففي الشكل (٦) يظهر أن النص الذي تم اختياره هو "كاتب" وأن "النصوص الفوقية" هو نص فوقي في هذه الشاشة . وعند ضغط المستخدم لزر "ادخل" يجري تنفيذ المنهاج المتعلقة بالنص الفوقي "كاتب" . وهكذا .



شكل (٦) : الشاشة كما تبدو بعد الإنتقال إلى نص فوقي

خوارزم اضافة نصوص فوقية في برامج كاتب :

١) يقوم المبرمج (المستخدم) بتعيين الكلمة (أو الجملة) المطلوبة لتكون كلمة (جملة) نص فوقي. يضع المبرمج هذه الكلمة (أو الجملة) بين قوسين مربعين ونجمتين [* كلمة أو جملة نص فوقي *].

٢) يقوم المبرمج بكتابة التفصيلات المطلوبة لكلمة النص الفوقي بشكل منهاج في "كاتب بعنوان يطابق

كلمة (أو جملة) النص الفوقي المطلوب . ومن الممكن أن يستخدم المبرمج نصوص فوقية أخرى ضمن المنهاج أيضا .

٣) يضع المبرمج أمر "عمق" في آخر سطر في الصفحة المطلوب اعتبارها صفحة نصوص فوقية . فيما يتم تنفيذ البرنامج المكتوب بلغة "كاتب" ، يقوم مترجم "كاتب" بعرض الكلمات ذات النص الفوقي بلون مختلف عن بقية النصوص ثم يظهر للمستخدم عدة خيارات :

أ. إختيار النص الفوقي التالي للنص الحالي (زر "القفزة")

ب. إختيار النص الفوقي السابق للنص الحالي (زر "أعلى-قفز")

ج. تنفيذ النص الفوقي (زر "ادخل")

د. الاستمرار في تنفيذ البرنامج بشكل مسلسل (زر "المساحة")

هـ. العودة للصفحة السابقة (زر "صفحة اعلى")

و. الخروج أو العودة للفقرات السابقة (زر "خروج")

ويتم عرض الازرار التي على المستخدم ضغطها لتنفيذ أي من هذه الاختيارات .

يقوم مترجم "كاتب" عند قراءته لامر "عمق" باعتبار الصفحة الحالية صفحة نصوص فوقية فيقوم بعرض الصفحة والنصوص الفوقية والاختيارات الممكنة للمستخدم كما هو موضح في المثال السابق . وتحتوي عقدة النص الفوقي في لغة "كاتب" على أي أمر من الأوامر (بدون تحديد) وبذلك يتم استخدام كل امكانيات "كاتب" عند كل عقدة . ومن الممكن كذلك عرض صورة أو صوت أو رسمة بالاضافة للنصوص العادية والفوقية .

ويمتاز هذا التطبيق للنصوص الفوقية بالميزات التالية :

١) سهولة عمل وثائق ذات نصوص فوقية بنفس السهولة التي يجري بها كتابة برامج لغة "كاتب" .

٢) لا يتطلب ذلك إضافة أية تركيبات خاصة معقدة للغة "كاتب" التي قرر ابتداء ان تكون لغة سهلة الاستخدام والتعلم من قبل المبرمجين (والمدرسين خاصة) .

٣) تمكن المبرمج من استخدام كافة أوامر لغة "كاتب" في عقد نصوص الفوقية وبالتالي يمكن عرض الصور والصوت والرسومات بالاضافة للنصوص العادية .

٤) يمكن تعديل البرامج التي كتبت سابقا بلغة "كاتب" بسهولة لتعمل بالنصوص الفوقية .

٧ – أدوات بناء النظم الخبيرة في لغة كاتب :

يتكون النظام الخبير (Expert System) من ثلاثة اجزاء رئيسية وهي قاعدة الحقائق ، قاعدة المعلومات ، وآلة الإستدلال . وتتبني قاعدة الحقائق من الحقائق الدائمة في الموضوع المعين بالاضافة للحقائق الخاصة بحل مسألة معينة [٢٦ ، ٢٧] . وتتكون قاعدة المعلومات من قوانين تمكن من الإستدلال باستخدام الحقائق وبذلك يمكن اضافة حقائق جديدة للقاعدة . ويحتوي كل قانون على طرف أيمن يتكون من "إذا" تم الشرط (أو الشروط) التي يجب توفرها ، ثم الطرف الايسر والذي يبدأ بـ "فإن" وهو يحدد ما يحصل اذا تحققت الشروط.

وتستخدم آلة الإستدلال (أو نظام التحكم) قاعدة المعلومات للتحقق من مشكلة ما بالاخذ بعين الاعتبار قاعدة الحقائق . فهي تتأكد من الجزء الايمن للقانون (طرف "إذا") للتأكد أن الشرط متحقق بالنظر الى قاعدة الحقائق . ويقرر نظام التحكم متى يتم استخدام القوانين ، كما أن عليه أن يحل أية تعارضات في حالة تحقق عدة قوانين . وهناك طريقتين للاستدلال ، الإستدلال الامامي والإستدلال الرجوعي . ويمثل الإستدلال الامامي بالإستدلال من أسفل لأعلى أو الإستدلال المدفوع بحادثه أو مرتبط بمعلومات . أما الإستدلال الرجوعي فيمثل بالإستدلال الهادف أو الإستدلال من أعلى لاسفل .

تقوم آلة الإستدلال في الإستدلال الامامي بسؤال ، ثم تأخذ اجابة المستخدم ، ثم تخرج آلة الإستدلال الامامي النتائج الممكن الحصول عليها من المعلومات المتوفرة . وإذا احتاج الامر تقوم آلة الإستدلال بعرض أسئلة أخرى حتى يتم حل الإستدلال بطريقة أو بأخرى [٢٧] . ولايبدأ النظام المستخدم للاستدلال الامامي بأي هدف محدد – أي ليس له قواعد معينة جزئية تبدأ منها نقطة الإستدلال .

أما في حالة الإستدلال الرجوعي فيقوم المستخدم باختيار نتيجة من قاعدة المعلومات . فيقوم النظام بالبحث الرجوعي لإيجاد الشروط التي يجب تحققها حتى يتم تحقيق النتيجة المطلوبة أو يظهر أن النتيجة قد تحققت [٢٧] . ويفحص النظام مجموعة محددة من القوانين التي يكون طرفها الايسر هو النتائج المطلوبة ثم يقوم النظام بفحص الجزء الايمن من القوانين للتأكد أي من هذه القوانين قد تم تحقيق شرطها . ومن هذا الفحص للقوانين يتم ايجاد الشروط الغير متحققة والتي تصبح نتائج ثانوية . فإذا كانت هذه النتائج الجزئية غير معروفة فيتم الاستفسار عنها لتحديد حالتها الراهنة أو اضافتها الى قائمةالشروط التي يتطلب تحقيقها .

ورغم ان معظم الانظمة استخدمت الإستدلال الرجوعي فإن الطريقتين للاستدلال يمكن استخدامها ما دام بالامكان الوصول للنتائج نفسها . وإختيار أي من الطريقتين في الإستدلال يعتمد على المشكلة التي يراد حلها . فبعض المشاكل يكون الهدف ثابتاً مقدماً ، وبهذا تكون طريقة الإستدلال الامامي هي الاولى بالاستخدام . وبعض الانظمة يستخدم الطريقتين بحيث تكمل احداهما الأخرى .

إن استخدام أي من طريقتي الإستدلال له تطبيقات كثيرة في أنظمة التعليم باستخدام الحاسب وبخاصة في نمذجة وتشخيص الطلبة . فنموذج الطالب هو نموذج بسيط لمعلومات وقدرات الطالب ، وفائدته هو تقييم أعمال الطالب واجاباته والاستجابة لها . فالمعلم عليه أن يعرف ما يعرف الطالب ، قدراته ، سرعة تعلمه استنادا لما سبق . كذلك عليه أن يعرف طرق التعليم التي يتجاوب معها الطالب بشكل أفضل ، وربما المواضيع التي عنده فيها خبرة ورغبة إضافة للموضوع الحالي . بالاستناد لهذه المعلومات فإن المعلم يستطيع أن يختار أنسب مستوى وطريقة العرض المناسب للطالب بحيث تعرض الموضوعات الجديدة بربطها بالموضوعات التي سبق للطالب معرفتها .

وكانت الطرق التقليدية في نمذجة الطالب تعتمد على المجموع العام للإجابات الصحيحة والخطأ . وللتأقلم مع الطالب كان يتم تغيير مسار خطوات التعليم بالاستناد لأجوبة الطالب أو إذا زاد مجموع اجاباته عن حد معين أو إذا زاد معدل اجاباته عن نسبة معينة . ان هذه النظم لا تتابع عملية زيادة علم الطالب إلا بشكل عام . لهذا فقد صممت نظم التعليم المعتمدة على الذكاء الصناعي بحيث تتجاوز هذا القصور. وفي هذه النظم يكون نموذج الطالب عبارة عن نموذج مفتوح مبني حول قاعدة معرفة [٢٨ . ٢٩] .

٧-١ تطبيق الإستدلال الامامي في لغة "كاتب" :

يتم استدعاء الإستدلال الامامي في لغة "كاتب" من خلال أمر خاص : " امامي" . والتركيب اللغوي لهذا الامر كما يلي :

امامي : قاعدة.ق

حيث قاعدة.ق هو اسم الملف المتضمن لقاعدة المعلومات .

وعند تنفيذ هذا الامر فإن مترجم "كاتب" يقوم بتنفيذ برنامج "كاتب" باستخدام وحدة البرمجة

المعتمدة على الإستدلال الامامي . وفي نهاية التنفيذ فإن قيمة الاستنتاج لقاعدة المعلومات يتم تعديلها حسب تنفيذ القواعد . ويستطيع معدّ الدروس أن يحصل على قيمة الاستنتاج كأي متغير .

ويتكون التركيب البياني لآلة الإستدلال الامامي من :

١. قائمة المتغيرات والتي تحتوي على أسماء وقيم المتغيرات .
٢. صف انتظار متغيرات الاستنتاج والتي تحفظ الإستدلالات التي هي نتيجة تطبيق الإستدلال الامامي لقواعد الذكاء الصناعي .
٣. مصفوفة تحتوي على جمل "اذا" والشروط والاستنتاجات .

أما خوارزم آلة الإستدلال الامامي فهي كما يلي:

١. يتم تحديد شروط كل القواعد .
٢. يتم وضع متغيرات الشروط في صف انتظار متغيرات الاستنتاج ويتم تحديد قيمتها في قائمة المتغيرات .
٣. يتم البحث في قائمة الشروط لاسم المتغير الذي يكون في مقدمة صف انتظار متغيرات الاستنتاج .
   أ) اذا وجد المتغير المطلوب فيتم تحديد قيمة الشرط المقابل له .
      فإذا تحقق الشرط فإن متغيرات طرف "اذن" يتم اعطاءها قيم جديدة .
      أما اذا لم يتحقق الشرط فإن متغيرات طرف "والا" يتم اعطاءها قيم جديدة.
   ب) يتم وضع المتغيرات التي تم تحديث قيمها في صف انتظار متغيرات الاستنتاج ويتم تحديد قيمها في قائمة المتغيرات .
٤. اذا لم يتبقى قواعد "إذا" اللاتي تحتوي على المتغير الذي في مقدمة صف انتظار متغيرات الاستنتاج ، فإنه يتم ازالة المتغير من صف الانتظار وإلا فيتم العودة لخطوة ٣.
٥. اذا وجدت متغيرات أخرى في صف انتظار متغيرات الاستنتاج فيتم العودة لخطوة ٣ والا فيتم الانتهاء.

٧-٢ تطبيق الإستدلال الرجوعي في "كاتب :

يتم استدعاء الإستدلال الرجوعي بواسطة استدعاء بأمر خاص " رجوعي" . والتركيب اللغوي للامر

كما يلي :

رجوعي : قاعدة.ق، $هدف، @نتيجة، مخرج

وهذا الامر له أربعة عوامل وهي:

قاعدة.ق وهو اسم الملف المحتوى على قاعدة المعلومات ،

هدف وهو متغير حرفي يعبّر عن الهدف المراد البحث عنه ،

نتيجة وهو متغير رقمي صحيح تعاد فيه نتيجة البحث في قاعدة المعلومات ،

مخرج وهو اسم لملف سيتم خزن جميع قوائم الشروط فيه اذا كانت القيمة المرجعة (نتيجة) ١ . وقد توضع كلمة "شاشة" أو "طابعة" بدلا من اسم الملف لتعني أن قوائم الشروط تحوّل الى الشاشة أو الطابعة على الترتيب اذا كانت القيمة المرجعة ١.

حينما يتم تنفيذ الأمر " رجوعي" فإن آلة الإستدلال تبحث عن الهدف المطلوب في الطرف الايسر من القواعد في قاعدة المعلومات . فإذا لم يوجد هذا الهدف في أي من القواعد (الطرف الايسر منها) فإن قيمة (-١) تعاد في المتغير "نتيجة" .

اما اذا وجد متغير "هدف" في واحد أو أكثر من القواعد فإن الشروط التي في الجزء الايمن من القاعدة يُنظر لها وكانها اهداف جديدة . ويستمر البحث عن هذه الاهداف الجديدة في الطرف الايسر من القواعد والشروط التي يجب توفرها لتتحقق . وتستمر هذه العملية حتى لا يبقى اهداف جديدة يلزم تحقيقها . و نتيجة هذا البحث هي قوائم للشروط التي يجب توفرها حتى يتم تحقيق الهدف المطلوب . ويتم حفظ القوائم في ملف أو عرضها على الشاشة أو طباعتها حسب رغبة المستخدم . بعد ذلك يقوم المستخدم بدراسة الشروط واختيار القائمة الانسب (إذا كانت هناك أكثر من قائمة) لتحقيق الهدف . ونتيجة هذا البحث قد تكون مايلي :

(١) قد تكون الشروط اللازمة لتحقيق الهدف محققة وبهذا يعيد النظام رقم صفر في متغير "نتيجة" ليدل على تحقق الهدف .

(٢) أما اذا لم تتحقق الشروط التي تحقق الهدف فإن رقم "١" يعاد في متغير "نتيجة" ، ويتم تعديل قيم المتغيرات بالقيم المطلوبة حتى يتحقق الهدف المطلوب .

ويتكون التركيب البياني للاستدلال الرجوعي مما يلي :

١. قائمة المتغيرات والتي تحتوي على اسماء وقيم المتغيرات .

٢. قائمة الأهداف ليتم حفظ الهدف الرئيسي والاهداف الفرعية .

٣. مصفوفة تحتوي على جمل "اذا" والشروط والاستثناءات .

أما خوارزم الإستدلال الرجوعي فهو كما يلي:

١. حدد الهدف المطلوب وضعه على قائمة الاهداف .

٢. ابحث عن الهدف المطلوب في الطرف الايمن من القواعد ، فإن لم يوجد الهدف في أي من القواعد (الطرف الايسر) فيتم اعادة رقم -١ في متغير "نتيجة" حتى يبين للمستخدم أنه لا يمكن تحقيق الهدف وينتهي الإستدلال . أما ان وجدت قاعدة أو أكثر (الطرف الايسر) فيها نفس الهدف ، فإنه يتم وضع الشروط لتلك القاعدة (أو القواعد) في قائمة الاهداف .

٢-أ إبحث عن كل هدف في قائمة الاهداف يكون فيه هذا الهدف نتيجة في القاعدة (الطرف الايسر).

٢-ب تصبح الشروط التي تم الحصول عليها في نقطة ٢-أ أهدافاً جديدة . يتم نسخ هذه القائمة الى عدد من القوائم (بقدر عدد القوائم التي تم ايجادها في نقطة ٢-أ) وإمسح القائمة الاصلية .

٢-ج لكل قاعدة تم ايجادها في ٢-أ توضع شروط كل قاعدة في القائمة الملائمة لها .

٢-د كرر ٢-أ الى ٢-ج حتى لا يبقى أي هدف جديد نتيجة في أي من القواعد .

٣. كرر خطوة ٢ حتى يتم استنفاذ كل القوائم .

٤. كرر ما يلي لكل القوائم التي تم ايجادها في الخطوات ١-٣.

٤-أ تأكد من تحقق الهدف الرئيسي والاهداف الجديدة ، فإذا تحقق الهدف امحه .

٤-ب امح أي تكرار في الاهداف .

٥. تأكد من عدم احتواء قائمة لقائمة أخرى فإن وجدت قائمة تحتوي قائمة أخرى فإنه يتم مسح القائمة الأكثر شروطا (الاطول) .

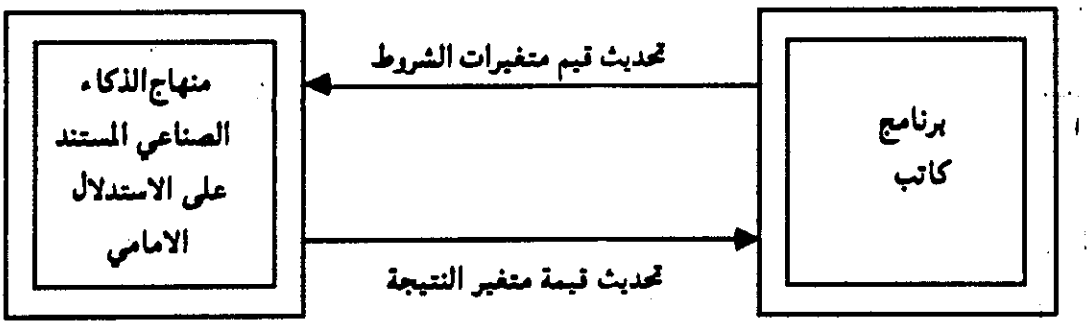
٧-٣ تحديد مستوى طالب باستخدام الإستدلال الامامي :

نعطي في هذا الفصل مثالاً على استخدام الإستدلال الأمامي في تحديد مستوى طالب بناء على مستوى أدائه في أحد الإمتحانات وبالاستنادإلى قاعدة المعلومات . إن نتيجة هذا الإستدلال يمكن الوصول اليها عن طريق برنامج "كاتب الذي يصل لهذه المعلومة كأي متغير في لغة "كاتب . وحسب مستوى الطالب يجري اعطاء الطالب مزيدا من الدروس أو مزيدا من مسائل التمرين لتحسين مستوى فهم الطالب للموضوع ثم إمتحانه مجددا . وتستمر هذه العملية الى أن يصل الطالب للمستوى المطلوب .

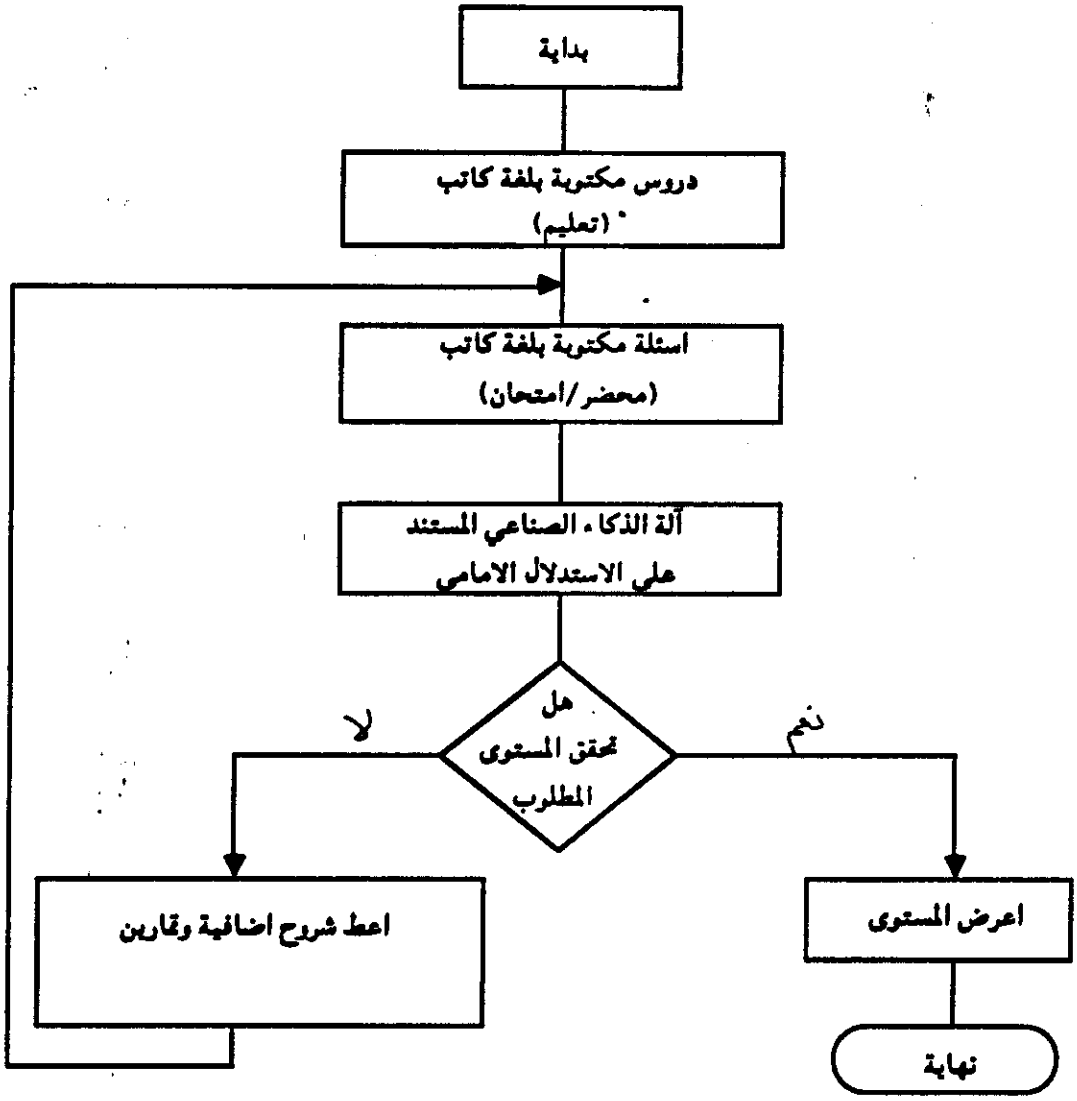
وفي هذا المثال نفترض أن موضوع المشكلة هو معرفة مستوى أداء الطالب في العمليات الرياضية التالية :

- جمع أرقام مكونة من خانة واحدة .
- جمع أرقام مكونة من أكثر من خانة .
- طرح أرقام مكونة من خانة واحدة .
- طرح أرقام مكونة من أكثر من خانة .

ويبين الشكل (٧) العلاقة بين برنامج "كاتب ومنهاج الذكاء الصناعي ذا الإستدلال الامامي ويبين الشكل (٨) خريطة خطوات العمل لايجاد مستوى الطالب .



شكل (٧) العلاقة بين برنامج مكتوب بلغة "كاتب ومنهاج الذكاء الصناعي المستند على الإستدلال الرجوعي



شكل (٨) : خريطة خطوات العمل لايجاد مستوى الطالب

ولتوضيح الخوارزم السابق سنفترض أن التعليم وفحص الطالب قد تم ، وأن نتيجة الطالب بعد هاتين الخطوتين هي كما يلي:

جمع-خانة = صح ، جمع-خانات = صح ، طرح-خانة = صح ، طرح-خانات = صح

بعد فحص الطالب يقوم برنامج "كاتب" باستدعاء منهاج الإستدلال الامامي بتنفيذ أمر "كاتب" :

أمامي : جمع-طرح.ق

حيث جمع-طرح.ق هو الملف المحتوي على القواعد الخاصة بالجمع والطرح ويبين الشكل (٩) هذا الملف .

هذا وقد تم اضافة رقم السطر لتسهيل الشرح عند عرض المثال .

١	اذا $جمع-خانة = خطا	و $جمع-خانات = خطا		اذن@جمع = ٠
٢	اذا $جمع-خانة = صح	و $جمع-خانات = خطا		اذن@جمع = ١
٣	اذا $جمع-خانة = خطا	و $جمع-خانات = صح		اذن@جمع = ٢
٤	اذا $جمع-خانة = صح	و $جمع-خانات = صح		اذن@جمع = ٤
٥	اذا $طرح-خانة = خطا	و $طرح-خانات = خطا		اذن@طرح = ٠
٦	اذا $طرح-خانة = صح	و $طرح-خانات = خطا		اذن@طرح = ١
٧	اذا $طرح-خانة = خطا	و $طرح-خانات = صح		اذن@طرح = ٢
٨	اذا $طرح-خانة = صح	و $طرح-خانات = صح		اذن@طرح = ٤
٩	اذا@جمع = ٠	و@طرح = ٠	اذن	$مستوى = ضعيف
١٠	اذا@جمع = ٠	و@طرح = ١	اذن	$مستوى = ضعيف
١١	اذا@جمع = ١	و@طرح = ٠	اذن	$مستوى = ضعيف
١٢	اذا@جمع = ١	و@طرح = ١	اذن	$مستوى = ضعيف
١٣	اذا@جمع = ٠	و@طرح = ٤	اذن	$مستوى = أقل من المتوسط
١٤	اذا@جمع = ٤	و@طرح = ٠	اذن	$مستوى = أقل من المتوسط
١٥	اذا@جمع = ١	و@طرح = ٤	اذن	$مستوى = متوسط
١٦	اذا@جمع = ٤	و@طرح = ١	اذن	$مستوى = متوسط
١٧	اذا@جمع = ٣	و@طرح = ٤	اذن	$مستوى = جيد
١٨	اذا@جمع = ٤	و@طرح = ٣	اذن	$مستوى = جيد
١٩	اذا@جمع = ٤	و@طرح = ٤	اذن	$مستوى = ممتاز

شكل (٩) : قاعدة المعلومات الخاصة بالجمع والطرح

فيقوم برنامج "كاتب باستدعاء منهاج الإستدلال الامامي الذي يقوم بتنفيذ الإستدلال باستخدام نتائج الطالب والقواعد المذكورة في ملف "جمع-طرح.ق" كما يلي:

١. الخطوة الاولى هي تحديد الشروط وهي:
جمع-خانة = صح ، جمع-خانات = صح ، طرح-خانة = صح ، طرح-خانات = صح

٢. يتم في هذه الخطوة وضع متغيرات الشروط في صف انتظار متغيرات النتائج والتي ستحتوي على متغيرات الشروط (جمع-خانة ، جمع-خانات ، طرح-خانة، طرح-خانات) . هذا وتوضع قيمة متغيرات الشروط في قائمة المتغيرات . يبين الشكل (١٠) قائمة المتغيرات ويبين الشكل (١١) صف انتظار متغيرات النتائج بعد تنفيذ هذه الخطوة .

٣. يجري البحث في شروط قاعدة المعلومات عن إسم المتغير الموجود في مقدمه صف انتظار متغيرات النتائج وهو جمع-خانة .

٤. من الملاحظ أنه تم العثور على جمع-خانة في الجملة الاولى في قاعدة المعلومات ، ولذا يتم تقييم شرط هذه القاعدة . وإذا لم يتحقق شرط هذه القاعدة ، يجري البحث عن "فإن" وحيث أنها غير موجودة لا يجري عمل شئ .

٥. يتم تكرار خطوة ٣ لكل القواعد (القوانين) وعليه فقد تم ايجاد جمع-خانة في القواعد ٤.٣.٢ في قاعدة المعلومات والشروط المقابلة لها خطأ ، خطأ ، صح على التتابع . لهذا يجري تعديل قيمة الطرف "اذن" في القاعدة الرابعة (حيث أن القواعد ٣.٢ خطأ) وعليه يجري تعديل قيمة جمع إلى ٤ ، وتوضع في صف انتظار متغيرات النتائج وتعدل قيمتها في قائمة المتغيرات . وحيث أنه جرى البحث في كل القواعد التي طرفها الايمن يحتوي على جمع-خانة فإنه يجري إزالة جمع-خانة من صف انتظار متغيرات النتائج . ويبين الشكل (١٢) قائمة المتغيرات والشكل (١٣) صف انتظار متغيرات النتائج .

٦. يجري تكرار تنفيذ الخطوات السابقة لكل المتغيرات في صف انتظار متغيرات النتائج . ويبين الشكل (١٤) قائمة المتغيرات بعد معالجة كافة المتغيرات في صف انتظار متغيرات النتائج .

ان قيمة النتيجة (التي تعبر عن مستوى الطالب) يجري تحديث قيمتها بعد معالجة كافة الشروط في آلة الإستدلال . وبعد الانتهاء من المعالجة يعود التنفيذ لبرنامج "كاتب حيث يقوم البرنامج بالوصول لنتيجة الطالب بسهولة كاي متغير في برنامج "كاتب . وهنا يمكن للمعلم المستخدم لنظام

"كاتب" أن يعطي الطالب توضيحات جديدة ، أمثلة أخرى ، أسئلة أخرى أو إنهاء الجلسة بعد أن عرف مستوى الطالب . يوجري تكرار هذه الخطوات حتى يصل الطالب للمستوى المطلوب .

اسماء المتغيرات	أخذت قيمة	القيمة
جمع-خانة	نعم	صح
جمع-خانات	نعم	صح
قيمة-جمع	لا	
طرح-خانة	نعم	صح
طرح-خانات	نعم	صح
قيمة-طرح	لا	
مستوى	لا	

شكل (١٠) قائمة المتغيرات بعد الخطوة ٢

جمع-خانة
جمع-خانات
طرح-خانة
طرح-خانات

شكل (١١) صف انتظار متغيرات النتائج بعد الخطوة ٢

اسماء المتغيرات	أخذت قيمة	القيمة
جمع-خانة	نعم	صح
جمع-خانات	نعم	صح
قيمة -جمع	نعم	صح
طرح-خانة	نعم	صح
طرح-خانات	لا	صح
قيمة-طرح	لا	
مستوى	لا	

شكل (١٢) قائمة المتغيرات بعد الخطوة

جمع-خانات
طرح-خانة
طرح-خانات

شكل (١٣) صف انتظار متغيرات النتائج بعد الخطوة

اسماء المتغيرات	أخذت قيمة	القيمة
جمع-خانة	نعم	صح
جمع-خانات	نعم	صح
قيمة-جمع	نعم	صح
طرح-خانة	نعم	صح
طرح-خانات	نعم	صح
قيمة-طرح	نعم	صح
مستوى	نعم	ممتاز

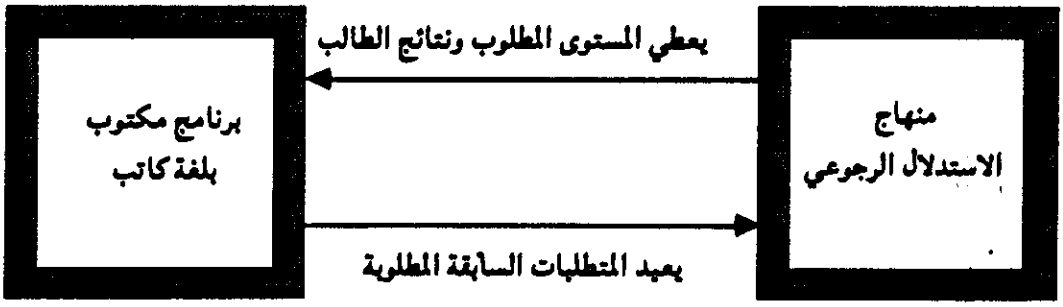
شكل (١٤) قائمة المتغيرات بعد العودة الى برنامج كانب من آلة الإستدلال

٧-٥ تشخيص الطلاب باستخدام الإستدلال الرجوعي :

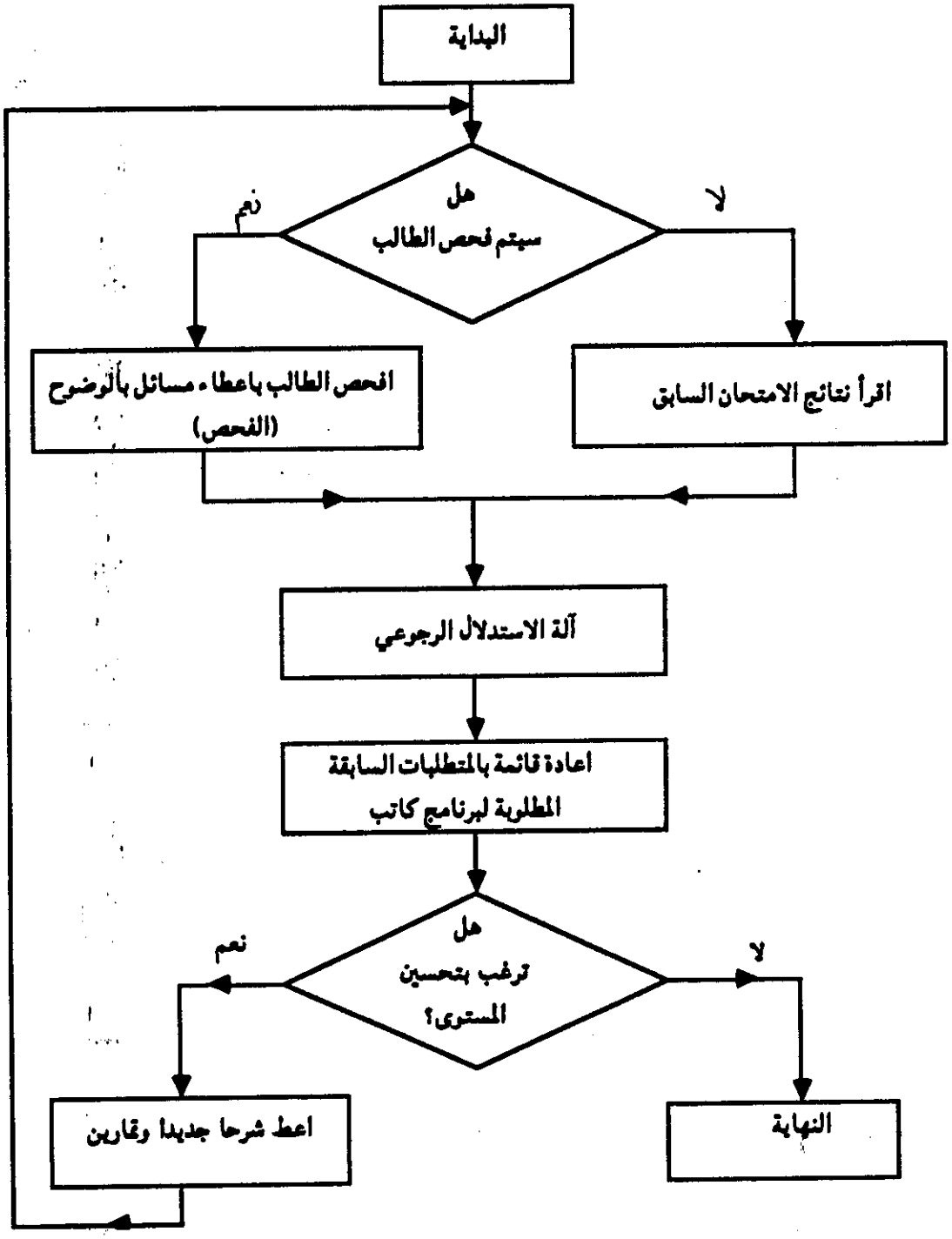
في المثال المذكور ادناه نبين كيفية استخدام الإستدلال الرجوعي لتحديد المتطلبات السابقة من المواضيع التي يتعين على الطالب اجتيازها . إن آلة الذكاء الصناعي المعتمدة على الإستدلال الرجوعي تعطي نتائج فحص الطالب بالاضافة الى المستوى المطلوب . فإذا حقق الطالب المستوى المطلوب فإن قيمة صفر تعاد في المتغير@نتيجة . أما إذا كان على الطالب أن يتعلم مواضيع جديدة حتى يصل للمستوى المطلوب فإن قيمة ١ تعاد في المتغير@نتيجة . وتعاد أيضا قوائم المواضيع التي على الطالب اجتيازها .

وقد يكون المستوى عبارة عن درجته ، فئته أو مستوى قدرته في لغة ما ...الخ . ويعطى منهاج الإستدلال الرجوعي المستوى المطلوب للطالب ، ونتائجه فيقوم المنهاج بمعالجة قاعدة المطلوب للاجابة على السؤال : فالمواضيع التي يجب على الطالب اجتيازها ليرتقي للمستوى المطلوب . وبعد الانتهاء تعاد قائمة بالمتطلبات السابقة المطلوبة ، واستنادا على ذلك فقد يقوم المعلم باعطاء الطالب دروس في مواضيع المتطلبات السابقة أو أمثلة وتمارين أو إمتحانات للارتقاء بالطالب للمستوى المطلوب . وتكرر هذه الخطوات حتى يصل الطالب للمستوى المطلوب .

ويبين الشكل (١٥) العلاقة بين برنامج كاتب ومنهاج الإستدلال الرجوعي كما يبين الشكل (١٦) خريطة خطوات العمل لايجاد المتطلبات السابقة .



شكل (١٥) : العلاقة بين برنامج كاتب ومنهاج الإستدلال الرجوعي



شكل (١٦) : خريطة خطوات العمل لايجاد المتطلبات السابقة

ولتوضيح خوارزم الإستدلال الرجوعي فقد جرى تتبع المثال التالي بإستخدام الإستدلال الرجوعي لايجاد المتطلبات السابقة للمستوى المطلوب . هذا وقد تم ترقيم الأسطر لتسهيل الاشارة اليها .

يقوم المعلم بفحص الطالب بموضوع القسمة والضرب باستخدام دروس مكتوبة بلغة "كاتب ثم امتحان الطالب ليعرف مدى قدرته او تؤخذ هذه النتائج من امتحان سابق ثم الاحتفاظ بنتائجه . لنفرض أنه بعد الفحص تبين أن الطالب عنده هذه النتائج :

ضرب-خانة = صح ، ضرب-خانات = خطا ، قسمة-خانة = صح ، قسمة-خانات = خطأ

ولنفرض أن المستوى المطلوب للطالب هو جيد .

يقوم برنامج "كاتب بعد الحصول على هذه النتائج والمستوى المطلوب باستدعاء منهاج الإستدلال الرجوعي كما يلي :

رجوعي : ضرب-طرح.ق، $مستوى، @نتيجة، شاشة

حيث ضرب-طرح.ق هو الملف المحتوى على قاعدة المعلومات كما هو موضح بالشكل (١٧)

$مستوى المستوى المطلوب للطالب وهو متغير حرفي .

نتيجة وهو متغير رقمي صحيح لاعادة الرقم المقابل لنتيجة الإستدلال

اعادة صفر يدل على أن الطالب لا يحتاج متطلبات سابقة

اعادة ١ يدل على أن الطالب يحتاج متطلبات سابقة .

شاشة أي تعرض المتطلبات السابقة على الشاشة

لايجاد المتطلبات السابقة تقوم الة الإستدلال الرجوعي بمعالجة القواعد كما يلي:

١. يقوم برنامج "كاتب باستدعاء منهاج الإستدلال الرجوعي الذي يقوم بمعالجة الإستدلال استنادا الى نتائج الطالب والمستوى المطلوب . يقوم المنهاج بوضع متغيرات النتائج في قائمة المتغيرات ووضع المستوى في قائمة المتطلبات السابقة كما يظهر في شكل (١٨) و (١٩) على الترتيب.

٢. تقوم آلة الإستدلال الرجوعي بأخذ قائمة من المتطلبات السابقة . وفي البداية توجد قائمة واحدة هي (مستوى = جيد) ثم تبحث على القواعد التي نتجتها مستوى = جيد . وقد وجد هذا

في القواعد ٢٠ ، ٢١ . وحيث أن هنالك قاعدتين تحققان الشرط يتم وضع نسختين كما هو موضح في الشكل (٢٠) (خطوة ٢ر٢ من الخوارزم) ثم توضح نتائج هذه القواعد (الموجودة في قواعده (٢٠ ، ٢١) في قائمة المتطلبات السابقة . ويبين الشكل (٢١) قائمة المتطلبات السابقة بعد تنفيذ هذه الخطوة .

٣. يتم تكرار خطوة ٢ لكل القوائم الموجودة في الشكل (٢١) .

٤. كرر خطوة ٣ حتى يتم استنفاذ كل القوائم . لقد تم استخلاص النتائج قيمة-ضرب=٢ وقيمة-قسمة=٣ بالاضافة الى قيمة-ضرب=٣ وقيمة-قسمة=٢ كما هو موضح بالشكل (٢١) من قواعد ٣ و ٧ . ويبين الشكل (٢٢) قوائم المتطلبات بعد نهاية هذه الخطوة .

٥. كرر ما يلي لكل القوائم التي تم إيجادها من الخطوات السابقة :

أ. تأكد اذا كانت هذه المتطلبات متحققة أم لا بالمقارنة مع قائمة المتغيرات. اذا كان أي متطلب متحقق ، امسح المتطلب . تظهر نتيجة الخطوة بالشكل (٢٣).

ب. امسح أي تكرار في قوائم المتطلبات وفي هذا المثال يتم ازالة أحد القوائم كما هو موضح في الشكل (٢٤) نتيجة للتكرار .

٦. افحص القوائم للتأكد من عدم احتواء قائمة لأخرى وحيث أنه لم يبق الا قائمة فلا تتم هذه الخطوة .

٧. وحيث أنه لا توجد قائمة خالية كما هو موضح بالشكل (٢٤) فإن قائمة المتطلبات هي الموجودة في الشكل (٢٤) . فالمتطلبات هي ضرب-خانات وقسمة-خانات . يتم عرض هذه المتطلبات على الشاشة .

٨. يتسلم برنامج "كاتب القيم المعدلة للمتغيرات من آلة الإستدلال . يقوم المعلم استنادا الى قائمة المتطلبات باعطاء الطالب شرحا جديدا أو امثلة جديدة أو امتحانا جديدا أو الانتهاء من هذه الجلسة بعد عرض قائمة المتطلبات السابقة لتحقيق المستوى المطلوب .

يتم تكرار الخطوات السابقة حتى يصل الطالب للمستوى المطلوب .

١	اذا $ضرب-خانة = خطا	و $ضرب-خانات = خطا	اذن@قيمة-ضرب = .
٢	اذا $ضرب-خانة = صح	و $ضرب-خانات = خطا	اذن@قيمة-ضرب = ١
٣	اذا $ضرب-خانة = خطا	و $ضرب-خانات = صح	اذن@قيمة-ضرب = ٢
٤	اذا $ضرب-خانة = صح	و $ ضرب-خانات = صح	اذن@قيمة-ضرب = ٣
٥	اذا $قسمة-خانة = خطا	و $قسمة-خانات = خطا	اذن@قيمة-قسمة = .
٦	اذا $قسمة-خانة = صح	و $قسمة-خانات = خطا	اذن@قيمة-قسمة = ١
٧	اذا $قسمة-خانة = خطا	و $قسمة-خانات = خطا	اذن@قيمة-قسمة = ٢
٨	اذا $قسمة-خانة = صح	و $قسمة-خانات = صح	اذن@قيمة-قسمة = ٣
٩	اذا@قيمة-ضرب = .	و@قيمة-قسمة = .	اذن $مستوى = ضعيف
١٠	اذا@قيمة-ضرب = .	و@قيمة-قسمة = ١	اذن $مستوى = ضعيف
١١	اذا@قيمة-ضرب = ١	و@قيمة-قسمة = .	اذن $مستوى = ضعيف
١٢	اذا@قيمة-ضرب = ١	و@قيمة-قسمة = ١	اذن $مستوى = ضعيف
١٣	اذا@قيمة-ضرب = .	و@قيمة-قسمة = ٢	اذن $مستوى = ضعيف
١٤	اذا@قيمة-ضرب = ٢	و@قيمة-قسمة = .	اذن $مستوى = ضعيف
١٥	اذا@قيمة-ضرب = ١	و@قيمة-قسمة = ٢	اذن $مستوى = مقبول
١٦	اذا@قيمة-ضرب = ٢	و@قيمة-قسمة = ١	اذن $مستوى = مقبول
١٧	اذا@قيمة-ضرب = ٢	و@قيمة-قسمة = ٢	اذن $مستوى = مقبول
١٨	اذا@قيمة-ضرب = ١	و@قيمة-قسمة = ٣	اذن $مستوى = مقبول
١٩	اذا@قيمة-ضرب = ٣	و@قيمة-قسمة = ١	اذن $مستوى = مقبول
٢٠	اذا@قيمة-ضرب = ٢	و@قيمة-قسمة = ٣	اذن $مستوى = جيد
٢١	اذا@قيمة-ضرب = ٣	و@قيمة-قسمة = ٢	اذن $مستوى = جيد
٢٢	اذا@قيمة-ضرب = ٣	و@قيمة-قسمة = ٣	اذن $مستوى = ممتاز

شكل (١٧) قاعدة المعلومات الخاصة بالضرب والقسمة

اسماء المتغيرات	أخذت قيمة	القيمة
ضرب-خانة	نعم	صح
ضرب-خانات	نعم	خطا
قيمة-ضرب	نعم	١
قسمة-خانة	نعم	صح
قسمة-خانات	نعم	خطا
قيمة-قسمة	نعم	١
مستوى	نعم	مقبول

شكل (١٨) قائمة المتغيرات بعد الخطوة الاولى

مستوى = جيد

شكل (١٩) قائمة المتطلبات السابقة

مستوى = جيد
مستوى = جيد

شكل (٢٠) قائمة المتطلبات السابقة بعد تحديد القواعد التي تحقق المستوى ويظهر الشكل نسختين احداها لقاعدة ٢ والاخرى لقاعدة ٢١

قيمة-ضرب = ٢	قيمة-قسمة = ٣
قيمة-ضرب = ٣	قيمة-قسمة = ٢

شكل (٢١) قائمة المتطلبات السابقة بعد تبديل الشرط

"مستوى = جيد" بالشروط التي تحقق المستوى المطلوب

ضرب-خانة = خطا	ضرب-خانات = صح	قسمة-خانة = صح	قسمة-خانة = صح
ضرب-خانة = صح	ضرب-خانات = صح	قسمة-خانة = خطا	قسمة-خانات = صح

شكل (٢٢) قائمة المتطلبات السابقة بعد استنفاد كل القواعد

ضرب-خانات = صح	قسمة-خانات = صح
ضرب-خانات = صح	قسمة-خانات = صح

شكل (٢٣) قائمة المتطلبات السابقة بعد ازالة المتطلبات المحققة

ضرب-خانات = صح	قسمة-خانات = صح

شكل (٢٤) قائمة المتطلبات السابقة بعد فحص التكرار

المراجع :

1- Alpert, D., and Bitzer, D.L. "Advances in Computer-Based Education," Science, vol. 167, 1976, pp. 1582-90.

2- Bitzer, D.L. "The Wide World of Computer-Based Education," in *Advances in Computers*, M.C. Yovits and M. Rubinof (eds.), vol. 15, Academic Press, 1976, pp. 239-83.

3- Romaniuk, E.W. *A Versatile Authoring Language for Teachers.* (PhD. Thesis). University of Alberta, Ednonton, Alberta, 1970.

4- Dean, P.M. "Computer-Assisted Instruction Authoring Systems," Educational Technology, vol. 18, no. 4, April 1978, pp. 20-23.

5- Nievergelt, J. "A Pragmatic Introduction to Courseware Design," IEEE Computer, vol. 13, Sept. 1980, pp. 7-21.

6- Starkweathers, J. *Nevada PILOT Reference Manual*, ELLIS Computing, U.S.A., 1982.

7- Kearsley, G. "Authoring Systems in Computer Based Education," Comm. of the ACM, vol. 25, no. 7, July 1982, pp. 429-437.

8- Park, O.C., Perez, R.S. and Seidel, R.J. "Intelligent CAI: Old Wine in New Bottles, or a New Vintage?," in *Artificial Intelligence and Instruction*, Kearsley, Ed. Reading, MA: Addison Wesley, 1987.

9- Carbonell, J.R. "AI in CAI: An Artificial Intelligence Approach to Computer Assisted Instruction." IEEE Trans. Man-Machine Sys., vol. 11, no. 4, Dec. 1970, pp. 190-202.

10- Koffman, E.B. and Blount, S.E. "Artificial Intelligence and Automatic Programming in CAI," Artificial Intelligence, vol. 6, 1975, pp. 215-234.

11- Koffman, E.B. and Perry, J.M. "A Model for Generative CAI and Concept Selection," Int. J. Man-Machine Stud., vol. 8, 1976, pp. 397-410.

12- Rickel, J.W. "Intelligent Computer-Aided Instruction: A Survey Organized Around System Components." IEEE Trans. on Systems, Man, and Cybernetics, vol. 19, no. 1, 1989, pp. 40-57.

13- Papert, S. *Mindstorms*. New York: Basic Books, 1980.

14- Freedman, R.S. and Rosenking, J.P. "Designing Computer-Based Training Systems: OBIE-1:KNOBE," IEEE Expert, Summer 1986, pp. 31-38.

15- Stevens, A., Collins, A. and Goldin, S. "Misconceptions in Students' Understanding." in *Intelligent Tutoring Systems*, Sleeman and Brown, Eds., Cambridge, MA: Academic Press, 1982, pp. 13-24.

16- Edwards, M. "The Mercedes Benz of Interactive Video," Hardcopy, vol. 14, no. 5, May 1985, pp. 74-80.

17- Brown, J.S., Burton, R.R., and deKleer, J. "Pedagogical, Natural Language

and Knowledge Engineering Techniques in SOPHIE I, II and III," in *Intelligent Tutoring Systems*, Steeman and Brown, Eds., Cambridge, MA: Academic Press, 1982, pp. 227-282.

18- Feigenbaum, E.A. and Barr, A. (eds). *The Handbook of Artificial Intelligence*. vol. 2, Los Altos, Ca: William Kaufmann, 1982.

19- Shneiderman, B., and Kearsley, G.P. *Hypertext Hands-On!*, Addison Wesley Publishing Co., 1988.

20- Smith, J.B., and Weiss, S.F. "Hypertext", Comm. of the ACM, vol. 31, no. 7, July 1988, pp. 816-9.

21- Rada, R. *Hypertext: from Text to Expertext*, McGraw-Hill Co., U.K., 1991.

22- Mandurah, M.M., and Azhari, I. "CATIB: An Arabic Authoring Language for CAI," Proc. of the 7th Saudi NCC, Riyadh, Jan. 1984, pp. 271-8.

23- Mandurah, M.M. and Dehlawi, F.M-A. *The Development of Computer-Assisted Instruction Systems in Arabic*. KACST Project #AR-5-100, Final Report, 1987.

24- Mandurah, M.M., and Mahmoud, S.A. *Development of An Advanced Arabic Authoring Language for CAI Applications*, KACST project #AR-11-42, Progress Reprot #2, KACST, Riyadh, April 1992.

25- Mandurah, M.M., and Mahmoud, S.A. *CATIB User Manual*, to be published.

26- Bonnet, A., J.P., and Truong-Ngoc, J.M. *Expert Systems*. Prentice Hall Int. (UK) Ltd., 1988.

27- Naylor, C. "How to Build an Inferencing Engine," in *Expert Systems: Principles and Case Studies*, Richard Forsyth, Ed., Chapman and Hall, UK, 1984.

28- Nawrocki, L.H. "Artificial Intelligence Applications to Maintenance Training," in *Artificial Intelligence and Instruction*, Kearsley, Ed., Reading, MA: Addison Wesley, 1987.

29- Pirolli, P.L. and Greeno, J.G. "The Problem Space of Instructional Design," in *Intelligent Tutoring Systems: Lessons Learned*, Psotka, Massey, and Mutter, Eds. Hillsdale, NJ: Lawrence Erlbaum Assoc., 1988.

ملحق أ

(١) أوامر الادخال والاخراج

اطبع	(Print)	لطباعة رسالة والانتقال للسطر التالي
اطبع-قف	(Printh)	لطباعة رسالة والتوقف
اطبع-وسط	(Printc)	لطباعة رسالة في وسط السطر والانتقال للسطر التالي
اطبع-وسط-قف	(Printch)	لطباعة رسالة في وسط السطر والتوقف
اكتب	(Type)	تعرض رسالة على الشاشة والانتقال للسطر التالي
اكتب-قف	(Typeh)	تعرض رسالة على الشاشة والتوقف
اكتب-وسط	(Typec)	تعرض رسالة على الشاشة في وسط السطر والانتقال للسطر التالي
اكتب-وسط-قف	(Typech)	تعرض رسالة على الشاشة في وسط السطر والتوقف .
خذ	(Accept)	لأخذ جواب المستخدم والانتقال للسطر التالي
خذ حرف	(Accepts)	لأخذ حرف من المستخدم والانتقال للسطر التالي
خذ قف	(Accepth)	لأخذ رسالة من المستخدم والتوقف
قائمة	(Menu)	تعرض قائمة من الاختيارات
عمق	(hyper)	لعرض شاشة العمق والاوامر الماضية بها واستدعاء برنامج معالجة العمق .
قل	(Say)	لقول رسالة باستخدام الصوت
ملاحظة	(Remark)	
تذييل	(Foot)	عرض رسالة في اسفل الشاشة واستقبال رد من المستخدم
طول-مدخل	(Input max)	يحدد أطول عدد من الحروف يمكن استقبالها من المستخدم .
صفحة	(Page)	يعرض رسالة أسفل الشاشة ويأخذ رد من المستخدم ثم ينتقل الى الصفحة السابقة أو التالية حسب رد المستخدم .

(٢) أوامر التحكم بالمسار

افعل ... حتى	(Do ... until)	كرر الاوامر بين "افعل" و"حتى" حتى يتحقق الشرط بعد حتى
نهاية	(End)	انهاء البرامج أو
طالما ... نهاية طالما	(While ... endwhile)	يكرر ما بعد الشرط من أوامر مادام الشرط متحقق
اذا ... فإن ... والا ... نهاية اذا	(If ... then ... else ... endif)	العبارة اذا فإن وإلا حيث اختبار الشرط فان تحقق يتم تنفيذ الاوامر بعد فإن وان لم يتحقق يت تنفيذ الاوامر عد والا .
اقصد	(Go to)	اقصد عنوان (انتقل الى العنوان المذكور)
انتظر	(Pause)	امر توقف لفترة محدودة .

تحويل حالة غادر خلاف ذلك نهاية تحويل : لاتخاذ قرارات متعددة فحسب قيمة ما بعد التحويل

(Switch, case, break, default, end swithc) يتم اختيار الحالة الملائمة وتنفيذ ثم تنقطع بـ "غادر" وان لم يتحق وجود حالة ملائمة يتم تنفيذ ما بعد خلاف ذلك.

مجموعة	(Group)	بداية مجموعة .
من .. نهاية من	(For .. end for)	كرر الاوامر حتى "نهاية من" بعدد المرات المعطاة
منهاج	(Subroutine)	استدعاء منهاج وتنفيذه
منهاج-عشوائي	(Subroutiner)	استدعاء منهاج عشوائيا تم تنفيذه .

(٣) أوامر التحكم بالشاشة

استعمل-نافذة	(usew)	اجعل النافذة المحددة في النافذة المستخدمة
اعكس-خلفية	(Reverse sides)	اعكس خلفية الشاشة
اغلق-نافذة	(Closew)	اغلق نافذة مفتوحة
افتح-نافذة	(Openw)	افتح النافذة
امسح-باقي	(Cleare)	امسح الى نهاية النافذة (الشاشة) الحالية
امسح-سطر	(Clearl)	امسح حتى نهاية السطر
امسح-شاشة	(Clears)	امسح النافذة (الشاشة) الحالية

حرك-مؤشر	(Cursorm)	حرك المؤشر الى الموضع المحدد
دحرج	(Rollscreen)	دحرج النافذة
عنوان-مؤشر	(Cursorp)	احصل على موضع المؤشر الحالي
قائمة-نوافذ	(Window list)	يعرض اسماء وارقام النوافذ المفتوحة
لون	(Setcolor)	غير لون النص إلى اللون المعطى
نسق	(Justify)	نسق المدخلات والمخرجات بين الهامشين المذكورين

(٤) أوامر معالجة حزم الحروف

الحق-حزمة	(Strcat)	زضف حزمة الى أخرى
انسخ-حزمة	(Str copy)	انسخ حزمة على أخرى
حزمة	(Str)	حول عدد صحيح الى حزمة
طابق	(Match)	طابق النماذج
طابق-اقفز	(Matchj)	طابق النماذج واقفز الى أمر "طابق" أو "طابق-اقفز" في حالة عدم المطابقة
طول	(Length)	يحسب طول حزمة
قارن-حزمة	(Str comp)	قارن حزمة بأخرى
قيمة	(Val)	تحويل حزمة الى عدد صحيح

(٥) الأوامر الرياضية والمنطقية

احسب	(Compute)	يحسب قيمة عبارة رياضية أو منطقية
اس	(Exp)	يحسب القيمة الاسية الطبيعية للأس المعطى
جا	(Sin)	احسب قيمة الجيب للزاوية المعطاة
جتا	(Cos)	احسب قيمة جيب التمام للزاوية المعطاة
جذر	(Sqrt)	احسب قيمة الجذر التربيعي للعدد المعطى
ظا	(Tan)	يحسب قيمة ظل الزاوية المعطاة
قارن	(Compare)	قارن عبارتين

لو	(Log)	يحسب القيمة اللوغاريثمية للاساس ١٠ للعدد المعطى
مربع	(Sqr)	يحسب قيمة العدد المعطى
مطلق	(Abs)	يحسب القيمة المطلقة للعدد المعطى

(٦) أوامر الرسم

ارسم	(Draw)	يرسم صورة مخزنة في ملف
اعرض-صورة	(Put image)	اعرض صورة ثنائية سبق تخزينها الى الشاشة
اعرض-عنصر	(Put pixel)	ارسم عنصراً في النقطة المحددة على الشاشة
اغلق-رسم	(Close graph)	يغلق حالة الرسم
افتح-رسم	(Open graph)	يفتح حالة الرسم
اكتب-رسم	(Write graph)	اكتب رسالة في حالة الرسم
امسح-رسم	(Clearg)	امسح الشاشة في حالة الرسم
بيضاوي	(Ellipse)	ارسم شكل بيضاوي
حجم-صورة	(Graphs)	احسب حجم الذاكرة اللازمة للصورة
خذ-صورة	(Get image)	اخزن صورة ثنائية للمساحة المحددة على الشاشة في الذاكرة
خذ-عنصر	(Get pixel)	خذ قيمة (لون) عنصر للنقطة المحددة على الشاشة
دائرة	(Circle)	ارسم دائرة على مركزها وطول نصف قطرها
عمود	(Bar)	يرسم عمود ذه بعد ثنائي
قوس	(Arc)	يرسم قوس دائري
مستطيل	(Rectang)	يرسم مستطيلا حسب الابعاد المعطاة
مستقيم	(Line)	يرسم مستقيما بين نقطتين معطاتين
مضلع	(Poly)	يرسم الحدود الخارجية لمضلع حسب المعلومات المعطاة

(٧) أوامر الملفات

اغلق-ملف	(Closef)	اغلق الملف المحدد أو جميع الملفات المفتوحة
افتح-ملف	(Openf)	افتح الملف المحدد للقراءة أو الكتابة

اقرأ-ملف (Readf) يقرأ من ملف

اكتب-ملف (Writef) يكتب في ملف

(٨) أوامر التحكم ببيئة كاتب

أمامي (Forward) يستدعي وحدة الذكاء الاصطناعي الامامي

تاريخ (Get date) اظهر التاريخ الحالي

حرر-ذاكرة (Free memory) يحرر مساحة من الذاكرة سبق حجزها

جدد-متغير (New var) امسح المتغيرات المحددة او كلها

خصص-ذاكرة (Get memory) احجز جزء من الذاكرة

رجوعي (Backward) يستدعي وحدة الذكاء الاصطناعي الرجوعي

زمن (Get time) يظهر الزمن الحالي

ملحق (ب)
التركيب اللغوي للغة كاتب

احسب : (@ صحيح=عبارةصحيحة| ٪حقيقي=عبارةحقيقية| $ حزمة=عبارةحزمة)

اذا : شرط فان
عمل : [عمل :] [عمل :]..
[والا :]
عمل : [عمل :] [عمل :]..
نهايةاذا :

ارسم : [@]س، [@]ص، [$]اسم-ملف
اس : (@ | ٪)اسم-متغير،[@ | ٪]عدد
استعمل-نافذة : [[@]عدد]

اطبع : [رسالة]
اطبع-ق : [رسالة]
اطبع-وسط : [رسالة]
اطبع-وسط-ق : [رسالة]

اعرض-صورة : [@]س١،[@]ص١،[عنوان]،[[@]عملية]
اعرض-عنصر : [@]س،[@]ص،[[@| $]لون]

اعكس-خلفية :
اغلق-رسم :
اغلق-ملف : [[$]اسم-ملف]

اغلق-نافذة : [[@]عدد]

افتح-رسم :

افتح-ملف : [[$]اسم-ملف]،[[$]حالة]

افتح-نافذة : [@ اسم-متغير]،[[$]عنوان]،[[@]عمود-م]،[[@]صف-م]،[[@]عمود-ع]، [[@]صف-ع]،[[@ | $]لون-ن]،[[@ | $]لون-خ]،[[@ | $]لون-١]

افعل :

عمل : [عمل :[]عمل :]...

حتى : شرط

اقرأ ملف : [$ اسم-ملف،($ | @ | ٪)اسم-متغير،[@ عائد]

اقصد : [(*]عنوان | #خ | # م)

اكتب : [رسالة]

اكتب-رسم : [@]س،[@]ص،[$]رسالة،[[@ | $]لون]

اكتب-قف : [رسالة]

اكتب-ملف : [$]اسم-ملف،رسالة

اكتب-وسط : [رسالة]

اكتب-وسط-قف : [رسالة]

الحق-حزمة : $ حزمة-هدف،[$]حزمة-مصدر

امامي : [$]اسم-ملف.

امح-نهائي :

امح-رسم :

امح-سطر :

امح-شاشة :

انتظر : [@]عدد

انسخ-حزمة : $ حزمة-هدف،[$]حزمة-مصدر