Advances in Information Sciences and Service Sciences
Volume 2, Number 2, June 2010

An Investigation into Influence Factor of Student Programming Grade Using Association Rule Mining

[1]Mohamad Farhan Mohamad Mohsin, [2]Mohd Helmy Abd Wahab, Mohd Fairuz Zaiyadi
[3]Norita Md Norwawi, [4]Cik Fazilah Hibadullah,
[1,4] College of Arts and Science, Universiti Utara Malaysia
farhan, cikfazilah, zfairuz@uum.edu.my
[3]Universti Sains Islam Malaysia, Faculty of Science and Technology
norita@usim.edu.my
[2]Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia
helmy@uthm.edu.my
doi: 10.4156/aiss.vol2.issue2.3

Abstract

Computer programming is one of the most essential skills which each graduate has to acquire. However, there are reports that they are unable to write a program well. Researches indicated there are many factors can affect student programming performance. Thus, the aim of this study is to investigate the significant factors that may influence students programming performance based information from previous student performance using data mining technique. Data mining is a data analysis technique that able to discover hidden knowledge in database. The programming dataset used in this study comprises information on the performance profile of Universiti Utara Malaysia students from 4 different bachelor programs that were Bachelor in Information Technology, Bachelor in Multimedia, Bachelor in Decision Science and Bachelor in Education specializing in IT of the November session year 2004/2005. They were required to enroll introductory programming subject as requirement to graduate. The dataset consisting of 419 records with 70 attributes were pre-processed and then mined using directed association rule mining algorithm namely Apriori. The result indicated that the student who has been exposed to programming prior to entering university and scored well in Mathematics and English subject during secondary Malaysian School Certificate examination were among strong indicators that contributes to good programming grades. This finding can be a guideline to the faculty to plan a teaching and learning program for new registered student.

Keyword: *Data mining, Computer programming, Programming, Association rules*

1. Introduction

Computer programming is a core skill in Information Technology (IT) related studies at most universities in the world which each student has to acquire. In job market, there are many vacancies offered which requires good programming skills as a basis. This is an opportunity for candidates with good programming background to seek computer related job such as programmer, system analyst, IT Professional and software engineer. However, graduates with low programming skill have less opportunity in software related job. Since the universities in Malaysia produced many IT graduates every year, the industry now has many potential candidates to be assigned as IT professionals. However, a wrong choice of graduates who do not meet the job requirements such as good programming skills or unsuitable IT personnel will be an expensive mistake. Lau (2002) reported that Malaysian IT students not only need to know programming language but also good analytical skills which they are lacking.

Learning to program is not an easy task to many students. They face difficulties when learning to program for the first time (Pillay & Vikash, 2005). Most students believe that computer programming is difficult and therefore they require more time to master the core concept (Bergin & Reilly, 2005). Bergin & Reilly also indicated that student perception on the programming subject might contribute to their final grade on the subject as well as high school calculus and science result, gender, and comfort level.

High failure rates were noticed among students in higher institution taking programming courses. Shukur et al. (2003) and Norwawi et al. (2005) reported that the failure rates in programming courses are about 30% to 40%. In solving problem through programming, the students are required to develop the abstract representation of the problems and express them in a logical structure and detailed realization in programming language which to them is complicated (Bennedsen & Caspersen, 2005). Naturally, programming requires thinking and analytical skill.

There have been a few studies in recent years on academic success in computer programming which lead to the assumption that intelligent student can write a program well. In contrast, there are students who are proficient at many other subjects sometimes failing to succeed in programming (Byrne a& Lyons, 2001). Thus, to assume that bright students can program is not necessarily true since there are cases where they fail to achieve success

Interest in the performance issue among programming students has long been studied among researchers. There are three trends in research examining predictors of programming success (Evans & Simkin, 1989). Prior to 1975, interest was in demographic background and secondary school achievement in mathematics and science subjects considered as the predictive variables. Then, in between 1975 to 1981, aptitude test scores were included and claimed as an effective predictors to measure innate cognitive mechanism such as ability to comprehend abstract concepts, recognize implicit information, interpret written problems and represent them into mathematical or symbolic form, summarize logic steps, decomposing problems into sub problems and following instructions in a procedures (Evans & Simkin, 1989; Goold & Rimmer, 2000). A seven-question exercise was adopted by Evans and Simkin (1989) while Goold and Rimmer (2000) used an eight problem-solving question in their evaluation. Similar instrument that measures the analytical skills, attention to detail, and ability to categorize and infer knowledge was adopted in this study as in Evans and Simkin (1989). In line with Chumra (1998), Pillay & Vikash (2005) found that language can be a barrier for student to develop programming skill if their first language not being same as the language of instruction.

Recent trend however, more interest is put in the non-technical predictor variables such as cognitive process that includes measures on behavioral habits, learning style, self-efficacy and/or personality. The focus of the recent study are on the relationship between non-technical factors and student's performance in introductory programming course, for instance personality type (Calitz *et al*., 1997; Haliburton *et al*., 1998), learning styles (Byrne & Lyons, 2001; Goold & Rimmer , 2000), personal motivation (Byrne & Lyons, 2001), self-efficacy (Ramalingam *et al*., 2004) and comfort level (Bergin & Reilly, 2005; Holden & Weeden, 2004). Further, self-efficacy, mental model and motivation can be drawn from previous experience or exposure to programming (Byrne & Lyons, 2001; Evans & Simkin, 1989; Ramalingam *et al*., 2004; Holden & Weeden, 2004).

In this study, personality type of students were attain from two psychometric profile test instrument which are Holland's Personality Classification as in Calitz *et al*. (1997) and Haliburton *et al*. (1998) and Color Personality (Muhamat Said, 2004). Color personality is new factor included as one of the predictor variable that is categorized into Gold, Blue, Green and Orange type. The Holland's personality has six classifications: Realistic, Artistic, Social, Investigative, Enterprising and Conventional as described in the Section 3.1.

As summarized in literature conducted by Norwawi *et al*. (2005), the high school mathematics grade, prior experience, cognitive ability, learning style, personality types, self efficacy, mental model, and gender are among the parameters for evaluating computer skills. The need of high-quality programming skills is increased in demand; therefore the learning ability of end user programming system is important. Research on learning barrier in programming courses has primarily focused on languages, overlooking potential barriers in the environment and human factors such as personality and aptitude. Therefore, the ability to predict an individual's potential to learn programming concept is important for many reasons Weinberg (1998). Norwawi *et al*. (2005) discovered students who have good background in Mathematics and English with Investigative type personality, have prior programming experience, and male are more likely to succeed in programming subject. They examined undergraduate Bachelor in Information Technology student's profile in order to identify the relationship between academic background, personality, and aptitude towards programming skill.

In this paper, data mining algorithm is used to understand the association relationship between variables. Data mining is a recent data analysis technique which can assist decision maker to extract hidden relationship from database. Data mining analysis has been applied in many domains such as business, medical, engineering, education and it has ability to provide additional guideline for future decision making (Mohsin & Abd Wahab, 2008). Ma *et al*. (2000) uses data mining algorithm for selection of students for remedial class using the O-level subjects results. Using the algorithm, students performing lower than the cut-off marks were recommended for the remedial classes. Ma *et al*. (2000) also claimed that the predictive model is much more precise. Since data mining can offers hidden knowledge which is hard to be seen through traditional data analysis, this paper aim to explore unique factors that contribute to the proficiency of students in computer programming using association rule mining approach.

Thus, to achieve that, a dataset of undergraduate student from Faculty of IT, UUM enrolled in second semester November session 2004/2005 obtained from the Academic Affairs department were mined using directed Apriori algorithm. The students were taught Java for the programming language. Apriori is one of association rule mining (AR) algorithm which searches the most frequent characteristics that occur together in database (Agrawal & Srikant, 1994). This dataset has been analyzed using statistical approach and decision tree (Norwawi *et al.*, 2005; Hibadullah & Norwawi, 2007). According to Bakar (2005), two different techniques will generate different sets of rules via knowledge although they are implemented to the same problem and produce comparable result. Therefore, the obtained knowledge using Apriori are then compared with their result.

This paper is organized as follows. Section 2 outlines the basic notion of AR. The model development of the study is discussed in section 3. The experiment and result will be presented in section 4 and final sections conclude this work.

2. Association Rule

In this section, the basic of association rule mining is discussed. Association rule mining or AR mining is the identification of frequent items that occur in a database of transaction. Each item (ij) in a transaction is an important feature that contributed to the computation of item set and generation of rules. Basically, let I = {i1,i2,…, im} be a set of item and D be a set of transactions, where each transaction T is a set of items such as that $T \subseteq I$. An AR is an implication of form $X \rightarrow Y$, where $X \subset I$, $Y \subset I$, and $X \cap Y = \emptyset$. The rule $X \rightarrow Y$ has support s in the transaction D if s% of transactions id D contain $X \cup Y$. The rule $X \rightarrow Y$ holds in the transaction with confidence c if c% of transaction in D that contain X also contain Y. AR mining's processes begin with searching for frequent item set with user-specified minimum support and later rules are contrasted by binding the frequent item with its values and class. Strong rules are defined as rules that have confidence more than the minimum confidence threshold.

3. Model Development

This study was divided into three phase. It started with data collection. After that preparing data for mining and ended with pattern extraction. The details of those activities were described in the next sub chapter. Figure 1 illustrates the data collection, data preparation for mining and pattern extraction phases of this study.

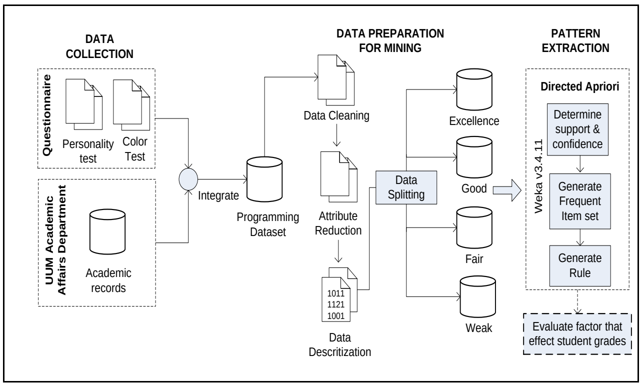

Figure 1. Model development of the study

3.1 Data Collection

There were two phases in the data collection. First, primary data was collected from two instruments randomly distributed to students who have taken introductory programming. Second, further data of the respondents on their previous academic background, demographic variable and grade in Introductory Programming course was obtained from the Academic Affairs Department of the university.

The two instruments were used to identify the personality type: Holland's Personality Classification and Color Personality. Students who have taken the subject were selected randomly from all semesters who were currently enrolled in year 2004/2005. 478 students responded to the exercise out of 1533 students. They represent about 31% of the population. Many students volunteer to participate as a subject. However upon examining the test, only 419 students response were used in this study. The students come from various program such as Bachelor in Information Technology (BIT), Bachelor in Multimedia (BMM), Bachelor in Decision Science (DEC) and Bachelor in Education specializing in IT (EDU) which enroll for introductory programming subject. Table 1 shows the distribution of students in terms of gender and program. There are 72% female students and 28% male students where about 75% students are from the BIT program.

Table 1. Distribution of students

		Program				
		BIT	**BMM**	**DEC**	**EDU**	**Total**
Gender	male	87	16	1	12	116
	female	226	33	3	41	303
Total		313	49	4	53	419

The obtain record is named as programming dataset. In general, the record includes the student grades in various programming subject particularly in introductory programming, demographic information such as gender, prior experience in programming, Malaysia Certificate of Education (SPM) grades in Bahasa Melayu, Mathematic and English. Results from the seven question aptitude test, and personality test are also incorporated. There are 419 records elaborated with 70 various types of attributes and a decision class. The result of introductory programming subject; TIA1013 is considered as decision class. Table 2 shows the grade classification according to the raw marks.

Table 2. Categories for programming performance based on TIA1013 subject

Category	Grades	Marks
Excellent	A, A-, B+	Above 70%
Good	B,B-	60%-70%
Fair	C, C+	50%-60%
Weak	D,D+,F	Less than 50%

Out of 419 students, 15.1% obtained excellent grade, 23.2% were recorded as good, 39% as fair and 22.7% in weak group. Preliminary observation on the raw dataset shows some attributes were not related to the study, certain values were missing and duplicate.

3.2 Data Preparation for Mining

During preprocessing task, all dataset were pre-processed where all unknown numeric attributes were replaced with mean value while max value for character attributes. Then, the data were discretized using boolean reasoning technique (Nguyen, 1998). The programming dataset contains large number of attributes (70 attributes) and some of them were not related to the study. Therefore only important attributes were selected for mining. During the selection process, nine attributes which stored student prior experience in programming languages namely Java, C++, C, Visual Basic, ASP, PHP, COBOL, Pascal, and Prolog were reclassified into three new groups based on the programming language characteristic. The new groups were Object, Structured, and Declarative. Figure 2 shows the reduction of nine attributes into Object, Structured, and Declarative.

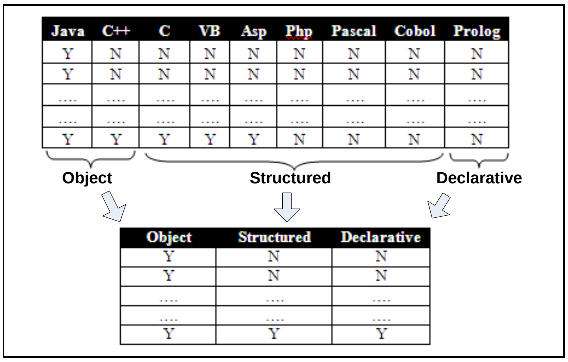

Figure 2. Attribute reduction on student experience attributes in particular programming language

Out of the 70 attributes, only 21 attributes were accepted for the next stage as depicted in Figure 3. Then, the records were split into 4 folds based on programming performance; excellent, good, fair, and weak. The fold represented the student programming proficiency in introductory programming course TIA1013. The output of this phase was a set of clean data.

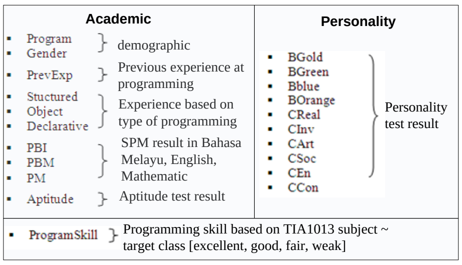

Figure 3. List of accepted attributes for mining

The attributes in Figure 3 represents the academics and personality information of the student. In academic group, it stores the demographic information of the students (*Program*, *Gender*), previous experience in programming before they enter the university (*PreExp*) and also types of programming they have learn (*Structured*, *Object, and Declarative*), their grade in Bahasa Melayu (*PBM*), Mathematic (*PM*) and English (*PBI*) subject during SPM and aptitude test result (*Aptitude*).

The *BGold*, *BGreen*, *BBlue*, and *BOrange* in personality group hold the scores of Color personality test while the *Creal, CInv, Cart, CSoc, CEn,* and *CCon* represent the result of Holland's personality test. Each score of the test represent the personality of the student. Table 3a and 3b elaborate the classification of both tests in detail.

Table 3a. Color Personality Category (Muhamat Said, 2004)

Color	Meaning
Gold (BGold)	Systematic, Responsible, Reliable, Conforming
Green (BGreen)	Patient, Curious, Philosophical, Complex, Cool, Knowledgeable
Blue (BBlue)	Pure, Cooperative, Unique, Creative
Orange (Borange)	Spontaneous, Brave, Adventurous, Skillful

Table 3b. Holland's Personality Type (Calitz *et al*., 1997; Haliburton *et al*., 1998)

Type	Characteristics
Investigative (CInv)	Curious, precise, unpopular, analytical and rational. They have scientific and mathematical ability. Prefer to work on their own in a research environment.
Realistic (CReal)	Asocial, Conforming, practical and persistent. Have mechanical ability Prefer to work with their hands, use tools, outdoors and caring for animals, crops and plants.
Artistic (CArt)	Impulsive, Disorganized, Original, Imaginative, Complicated, Creative individuals
Social (CSoc)	Emphatic, warm, kind, patient and helpful. Prefer to teach and help others.
Enterprising (CEn)	Dominant, adventure, self-confident, talkative and energetic, persuasive
Conventional (CCon)	Conforming, ordering, persistent, practical and unimaginative. Like working with numbers Prefer routine and predetermined instructions in a work environment

3.3 Pattern Extraction

During pattern extraction phase, AR algorithm called Apriori in WEKA data analysis tool was chosen as a pattern extraction tool (Witten & Frank, 2005). Since Apriori run only on nominal data type, all numeric values were transformed into nominal. Then, each fold was presented to Apriori algorithm and during mining, the length of frequent item set, support, and confidence value of each itemset was recorded. In this study, the minimum support value was set differently in each fold due to the number of cases in each fold was different. The mining output of each fold was then compared. Figure 1 illustrates the pattern extraction phases of this study.

This section reports the finding of the study. During experiment, different minimum support setting was applied to each fold yet minimum confidence was equally set to 90% to each fold. Theoretically, both values are importance in AR mining because the number of frequent item set to be generated will be based on minimum support value while the confidence value will filter only the quality rules. If the value is set too high, there is possibility no interesting rule can be found and yet too many patterns will be generated if lower threshold value is used (Liu et al., 1998). Table 3 shows the setting of the minimum support (*Sp*) and confidence (*Cf*) value and the quantity (*Qtty*) of data in each fold. Besides that, the maximum number of rule can be generated by Apriori was limited to 500 rules.

Table 4. Minimum support and confidence value

Fold	Qtty	Sp (%)	Cf (%)
excellent	61	30	90
good	99	40	
fair	164	50	
weak	97	40	

Table 5 summarizes the quantity of the knowledge which was mined from programming dataset. From the Table, different amount of item set and rule had been generated by Apriori. The *L1*, *L2*, *L3*, *L4*, and *L5* in item set columns shows the number of the most frequent attribute appeared together in programming data set. For example, *L3* in excellent group, there are 41 unique item set which each of them comprise three frequent factors. While the total number of the frequent item set in excellent group is recorded as 115. The last column of Table 4 represents the number of the most quality rules generated from L.

Table 5. The quantity of the item set and rule mined from programming dataset

Fold	Item set (*L*)						Rule
	L1-L5	*L1*	*L2*	*L3*	*L4*	*L5*	
Excellent	115	14	39	41	18	3	201
Good	124	15	44	45	18	2	155
Fair	110	14	35	35	19	7	195
Weak	90	9	23	36	16	6	203

The results were further analyzed. The experiment was focused on the relationship between academic factor and personality characteristic towards programming performance. Figure 2 lists the attributes according to academic factor and personality characteristics. To achieve that, we focused at the frequent item set in each group particularly excellent and weak. Table 6 represents a sample of the most frequent item set for excellent and weak group.

Table 6. A sample of the most frequent item set for excellent and weak group

Excellent
PrevExp=Y Stuctured=Y PM=A 19 Program=bit PrevExp=Y Stuctured=Y 22 PrevExp=Y Stuctured=Y PM=A 19 Program=bit PrevExp=Y Stuctured=Y Declarative=N 21 PrevExp=N Object=N Declarative=N PM=A 19 PrevExp=Y Stuctured=Y Object=Y Declarative=N 19 PrevExp=Y Stuctured=Y Declarative=N PM=A 18 PrevExp=N Stuctured=N Object=N Declarative=N PM=A 19
Weak
Stuctured=N Object=N PBI=D 38 Stuctured=N Declarative=N PBI=D 39 Object=N Declarative=N PBI=D 40 Object=N Declarative=N PBM=B 41 Program=bit Gender=p PrevExp=N Stuctured=N Object=N Declarative=N 54 Program=bit PrevExp=N Stuctured=N Object=N Declarative=N PBI=D 33 Program=bit PrevExp=N Stuctured=N Object=N Declarative=N PBM=B 34 Program=bit PrevExp=N Stuctured=N Object=N Declarative=N PM=C 30 Gender=p PrevExp=N Stuctured=N Object=N Declarative=N PBI=D 29

From the analysis on frequent item set, it shows that student with programming background particularly in structured programming has advantage to obtain a good grade at programming. However in certain item set, there were also students who do not learn programming before they enter the university but can excel in computer programming subject. Furthermore, the study also reveals that student with good grades in Mathematic and English (at least B grade) during SPM can scores well in programming subject. Other finding is in term of gender which is in agreement with Narwawi *et al*. (2005) whereby programming skill of male student is better compared to female. Besides that, students from BIT program were performed better than other programs.

The relationship between personality factors towards programming grades was also investigated. However, no item sets were generated related to personality characteristic. Therefore, we conclude that Apriori is not a suitable technique to mine personality characteristic information. This might be due to the fact that Apriori depends on attribute value's frequency during analysis while each student personality attributes holds different value which were lower than the minimum support condition. Formerly, Hibadullah et al. (2007) investigated the same dataset using decision tree and they found that student with investigative personality has better potential in writing a good program. Table 7 summarizes the findings of this study and its comparison with previous study on same dataset.

Table 7. Comparison with previous study

Factors	**Study/Approach**		
	Norwawi et al. (2005) / *Statistic approach*	Hibadullah et al. (2007) / *Decision Tree approach*	This study / *Apriori approach*
Academic	⌂ Good background in English, Mathematics in SPM ⌂ Previous exposure to programming	⌂ Good background in Mathematics in SPM	⌂ Good background in English, Mathematics in SPM at least B grade ⌂ Previous exposure to programming particularly structured programming. However it is not compulsory
Demographic	⌂ Male student scores batter	-	⌂ Male student scores batter
Personality	⌂ Average cognitive ability ⌂ Own investigative type of personality	⌂ Good cognitive ability ⌂ Own investigative type of personality	-

From the frequent item set via frequent factor that influence programming performance, Apriori algorithm then had generated many rules Table 8 shows a sample of the quality rules (where *Cf* > 90%) in excellent group.

Table 8. A sample of quality rules in excellent group.

Excellent
▪ PrevExp=Y Object=Y Declarative=N 19 ==> Stuctured=Y 19 conf:(1)
▪ Object=Y Declarative=N 19 ==> PrevExp=Y Stuctured=Y 19 conf:(1)
▪ Program=bit Stuctured=N 22 ==> PrevExp=N Object=N Declarative=N 20 conf:(0.91)
▪ Gender=l 21 ==> Program=bit 19 conf:(0.9)
▪ Gender=l Declarative=N 21 ==> Program=bit 19 conf:(0.9)
▪ Gender=l 21 ==> Program=bit Declarative=N 19 conf:(0.9)
▪ PrevExp=Y 31 ==> Stuctured=Y Declarative=N 28 conf:(0.9)
▪ PrevExp=Y PM=A 20 ==> Stuctured=Y Declarative=N 18 conf:(0.9)
▪ PrevExp=Y Stuctured=Y Declarative=N PM=A 18 ==> Program=bit conf:(0.9)

Many interpretations can be derived from the rules. For example, the last rule from Table 8 indicates that the excellent group student has prior experience in programming particularly in structured programming, obtained a good grade in Mathematic during SPM and there are 90% confidence they are from BIT program.

4. Conclusions

In this paper, a dataset of undergraduate student from Faculty of IT, UUM was mined using directed Apriori algorithm; one of the association rule mining algorithm. This dataset has been previously analyzed by several researchers and this paper was aimed to identify other unique characteristics using association rule technique. The experiment focused the relationship between academic factor and personality characteristic towards programming performance. The findings indicated that student prior experience in programming before university can contributes to a good grade. However, it is not a necessary condition since some students who do not have programming experience also can excel in programming subject. Furthermore, sex particularly male student who obtained a good grade in Mathematic and English during SPM also had been identified as a critical factor to master programming. Additionally, this study was unable to identify the relationship between personality characteristic and programming grade.

As conclusion, several preventive measures can be taken by the faculty mainly to the new registered student who do not have any experience in programming and do not perform well in Mathematic and English subject during SPM. Besides that, other unidentified factor can also contributes to student achievement in programming such as learning environment, motivation, learning facilities, and instructor ability which need to be considered by the faculty.

5. References

[1] Agrawal, R. and Srikant, R. (1994). "Fast algorithm for mining association rules.", Proc. Int. Conf. Very Large Databases, pp. 487-449.
[2] Bakar, A.A.(2005). "Propositional satisfiability method in rough set classification modeling for data mining", PHd Thesis. Universiti Putra Malaysia.
[3] Bennedsen, J. and Caspersen, M.E. (2005). "An Investigation of Potential Success Factors for an Introductory Model-Driven Programming Course". In *Proceedings of International Conference on Education Research (ICER05).* USA, ACM pp155-161.
[4] Bergin, S. and Reilly, R. (2005). "Programming: Factors that Influence Success", *Proceedings SIGCSE'05. Feb 23-27. Missourri, US.* pp 411- 415
[5] Byrne P. and Lyons, G. (2001). "The Effect of Student Attributes on Success in Programming.", ACM SIGSE Bulletin of the Proceedings of the 6th Annual Conference on Innovation and Technology in Computer Science Education 33(3) pp49-52.
[6] Calitz, A.P., Watson, M.B. and de Kock, G de V. (1997). "Identification and Selection of Successful Future IT Personnel in a Changing Technological and Business Environment". In Proceedings of the 1997 ACM SIGCPR Conference on Computer Personnel Research. California, USA. pp 31-35.
[7] Chumra, G.A. (1998). What Abilities are Necessary for Success in Computer Science? In SIGCSE Bulletin-inroads, Vol (30), No.4, pp55a-58a. ACM Press
[8] Evans, G.E. and Simkin, M.G. (1989). "What Best Predicts Computer Proficiency". Communications of the ACM, 32 (11). pp 1322-1327.
[9] Goold, A. and Rimmer, (2000). R. Factors Affecting Performance in First Year Computing. ISIGCSE Bulletin. 32(2). Pp 39-43.
[10] Haliburton, W., Thweat, M. and Wahl, N.J. (1998). "Gender Differences in Persdonality Components of Computer Science Students: A test of Holland's Congruence Hypothesis". In Proceedings of the 1997 ACM SIGSCE 98, Atlanta, USA. pp 77-81.
[11] Hibadullah, C.F., and Norwawi, M.N. (2007). "Classification of student's performance in programming course using decision tree". The Fifth International Conference on Information Technology in Asia, Kuching, pp 315-317.
[12] Holden, E. and Weeden, E. (2004). "The Experience Factor in Early Programming Education". In ACM Proceedings of the SIGITE'04. Oct. 28-30, Utah, USA. pp 211-218.
[13] Lau, L. (30 November 2002). Our Generation of IT Talents: The 'Unemployed' and the 'Unemployable'. Straits Times Singapore. Retrieved 30/05/2205 from http://www.usj.com.my/bulletin/.
[14] Liu, B., Hsu, W. and Ma, Y. (1998). "Integrating classification and association rule mining", Proc. Int. Conf. on Knowledge Discovery and Data Mining, pp. 487-489.
[15] Ma, Y., Liu, B., Wong, C.K., Yu, P.S., and Lee, S.M. (2000). "Targeting the Right Students Using Data Mining". In Proceedings of the Knowledge and Data Discovery (KDD2000), Boston, USA. Pp457-464.

[16] Mohsin, M.F. and Abd Wahab, M.H. (2008). "Comparing knowledge quality in rough classifier and decision tree classifier." Proceeding of 3rd IEEE International Symposium of Information Technology (ITSIM08), Kuala Lumpur, pp 1109-1114.
[17] Muhamat Said, M. (2004). "Studi Smart". Kuala Lumpur: Percetakan Cergas, 2004.
[18] Pillay, N. and Vikash, R.J. (2005). An Investigation into Student Characteristic Affecting Novice Programming Performance. In SIGCSE Bulletin-inroads, Vol (37), No.4, pp107-110. ACM Press
[19] Nguyen, H.S. (1998). "Descretization problem for rough set methods". Proc of First Int. Conf. on Rough Set and Current Trend in Computing, pp. 545-552.
[20] Norwawi, N.M., Hibadullah, C.F., and Osman, J. (2005). "Factors Affecting Performance in Introductory Programming". [CDROM]. In Proceedings of the International Conference on Qualitative Sciences and Its Applications (ICOQSIA).
[21] Ramalingam, V., LaBelle, D. and Wiedenbeck, S. (2004). "Self-Efficacy and Mental Models in Learning to Program". In ACM Proceedings of ITICSE'04 June 28-30. Leeds. UK. pp 171-175.
[22] Shukur, Z., Alias, M., Hanawi, S.A. and Arshad, A.(2003). "Faktor-faktor Kegagalan: Pandangan pelajar Yang mengulang Kursus Pengaturcaraan C", Paper presented in Bengkel Sains Pengaturcaraan (ATUR03), Kuala Lumpur.
[23] Weinberg, G. M. (1998). "The psychology of computer programming (silver anniversary edition). " New York: Dorsey House Publishing.
[24] Witten, I.H. and Frank, E. (2005). Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann: San Francisco.