### **Automated Scheduling System for FCSIT**

Ooi Ling Sern (Matric No. : WEK 98084)

Supervisor: Dr. Syed Malek Fakar Duani

Moderator: Assoc. Prof. Dr. Roziati Zainuddin

Faculty of Computer Science and Information Technology University of Malaya

Session 2001/2002

### Abstract

The final year thesis project as entitle, "Automated Scheduling System for FCSIT", is required o build a scheduling system for Faculty of Computer Science and Information Technology (FCSIT).

The objective of this project is to improve the current scheduling process from a manual system to a computerized system. At the end of the project, an automated scheduling system should be produce. This system also should be generic for most timetabling in school or any higher learning institutions.

i

# Acknowledgement

Firstly, I would like to grasp this opportunity to express my deepest gratitude and thank to my supervisor, Dr. Syed Malek Fakar Duani, for his excellent guidance and patience throughout this thesis project. His ideas and advice does contribute a lot to this project.

Next, my sincere appreciation is forwarded to Assoc. Prof. Dr. Roziati Zainuddin for being my moderator.

At the same time, I would like to thank all my friends who have directly or indirectly assisted and guided me during this short period.

mignite 4.1	List of Figures	
Chapter 1		
Figure 1.1	Issues in Schedule Construction.	6
Chapter 2		
Chapter #	Course information entries window.	
Figure 2.1	A heuristic scheduling framework.	13
Chapter 3		
Figure 3.1	Rule-based model.	29
Figure 3.2	Forward-chaining inference process.	30
Figure 3.3	Backward-chaining process.	32
Figure 3.4	The algorithm for course scheduling system.	41
Figure 3.5	Context level diagram for the automated scheduling	
	system.	43.
Figure 3.6	Diagram of the automated scheduling system.	44
Figure 3.7	Diagram of process on the course description.	45
Figure 3.8	Diagram of process on the room description.	45
Figure 3.9	Diagram of process on the lecturer description.	46
Figure 3.10	Diagram of process on the session description.	46

### Chapter 4

Major system design for automated scheduling system.	48
Sub-modules in course entries modules.	48
Sub-modules in room entries modules.	49
Sub-modules in lecturer entries modules.	49
Sub-modules in schedulable time entries modules.	49
Course information entries window.	51
Room information entries window.	52
Lecturer information entries window.	53
View courses information window.	54
View rooms information window.	55
View lecturers information window.	56
View students' schedule window.	57
View lecturers' schedule window.	58
View rooms' schedule window.	59
	Major system design for automated scheduling system. Sub-modules in course entries modules. Sub-modules in room entries modules. Sub-modules in lecturer entries modules. Sub-modules in schedulable time entries modules. Course information entries window. Room information entries window. Lecturer information entries window. View courses information window. View rooms information window. View lecturers information window. View students' schedule window. View lecturers' schedule window. View rooms' schedule window.

# List of Tables **Chapter 1** Table 1.1 The list f scheduling terms and their meanings. 7 **Chapter 3** Table 3.1 The basic symbols used in the data flow diagram. 42 Chapter 1 Software used and descriptions. Table 5.1 62

Literstory Review

Lt. Automated to, Interactive Scheduling

2.2 Solution Approaches

2.2.1. Meteristics Algorithms

2.2. Evolutionary Algorithms

2.2.3 Genetic Abgunithmy

	2.2.5	Table of Contents	15
		Network Flow Techniques	
Abstract		i	
Ack	nowledg	gement	ii
List	of Figur	res	iii
List	of Table	es	v
24			
Cha	pter 1	Introduction	1
1.1	Projec	ct Definition	2
	1.1.1	Intelligent System	2
	1.1.2	Definitions for "Schedule"	3
	1.1.3	Course Scheduling System	3
1.2	The C	Course Scheduling Problems	5
	1.2.1	Course Scheduling at FCSIT	5
	1.2.2	Issues in Course Scheduling	6
	1.2.3	Scheduling Terms	7
1.3 Objectives		tives	8
1.4	Scope		9
Chap	oter 2	Literature Review	10
2.1	Autom	nated vs. Interactive Scheduling	11
2.2	Solutio	Solution Approaches	
	2.2.1	Heuristics Algorithms	12
	2.2.2	Evolutionary Algorithms	14
	2.2.3	Genetic Algorithms	14

	2.2.4	Graph Coloring	15
	2.2.5	Memetic Algorithms	15
	2.2.6	Network Flow Techniques	16
	2.2.7	Simulated Annealing	17
	2.2.8	Tabu Search	17
	2.2.9	Constraint Logic Programming	18
2.3	Case	Studies	20
2.4	Progr	amming Language	23
	2.4.1	What is Prolog?	23
	2.4.2	Why Prolog?	23
	2.4.3	Features of the Software	24
		2.4.3.1 Visual Development of GUI Applications	24
		2.4.3.2 A Wealth of Facilities	24
2.5	Datab	ase	25
	2.5.1	What is Microsoft Access?	25
	2.5.2	Why Microsoft Access?	25
Chap	oter 3	Methodology	26
3.1	Constr	raints	27
	3.1.1	Constraints on Course Scheduling	27
		3.1.1.1 Lecturer	27
		3.1.1.2 Course	27
		3.1.1.3 Time	27
		3.1.1.4 Room	28
	in the		

3.2	Rule-Based System 2		
	3.2.1	Forward Chaining	29
		3.2.1.1 Advantages of Forward Chaining	31
		3.2.1.2 Disadvantages of Forward Chaining	31
	3.2.2	Backward Chaining	31
		3.2.2.1 Advantages of Backward Chaining	33
		3.2.2.2 Disadvantages of Backward Chaining	33
3.3	Desig	ning Backward Chaining System	34
	3.3.1	Combination of Heuristics Algorithms and Backward	
		Chaining	34
	3.3.2	Defining the Problems	35
	3.3.3	Defining the Goals	36
	3.3.4	Defining the Goal Rules	37
	3.3.5	Expanding the System	38
		3.3.5.1 Course Rules	38
		3.3.5.2 Lab Course Rules	39
3.4	Algori	ithms for Course Scheduling System	40
3.5	Proce	sses and Flows in the System	42
Chap	ter 4	System Design	47
4.1	Auton	nated Scheduling System Design	48
4.2	Datab	ase Design	50
4.3	User I	nterface Design	50
	4.3.1	Course Information Entries Window Design	51
	4.3.2	Room Information Entries Window Design	52

	4.3.3	Lecturer Information Entries Window Design	53
	4.3.4	View Courses Information Window Design	54
	4.3.5	View Rooms Information Window Design	55
	4.3.6	View Lecturers Information Window Design	56
	4.3.7	View Students' Schedule Window Design	57
	4.3.8	View Lecturers' Schedule Window Design	58
	4.3.9	View Rooms' Schedule Window Design	59
Chap	oter 5	System Implementation and Testing	60
5.1	Devel	oping Environment	61
	5.1.1	Hardware	61
	5.1.2	Software	62
5.2	5.2 System Development		63
5.3	.3 Database Development		64
5.4	Testin	g	65
	5.4.1	Module Testing	65
	5.4.2	Integration Testing	66
	5.4.3	Function Testing	67
	5.4.4	Other Testing	67
Chap	ter 6	System Evaluation and Conclusion	68
6.1	System	n Evaluation	69
	6.1.1	System Strength	69
	6.1.2	System Limitations	70
6.2	Future	Enhancements	71

6.3	Problem Encountered	73
6.4	Knowledge Gained	74
6.5	Conclusions	75
Refe	rences	76
User Manual		80
Appendix		101

Chapter



### **1.1 Project Definition**

Currently, most of the schools or universities still used the traditional method, which is manually to do the scheduling job. This method not only waste time but also may be cannot do it completely. So, with this project, we will save a lot of time and easily do the scheduling job.

This project will assist the schools or universities administration to do the scheduling. This project is called Automated Scheduling System for FCSIT.

#### 1.1.1 Intelligent System

Intelligence is defined as the capacity and ability to think, learn, reason and apply knowledge [5]. It refers to the ability to think and understand instead of doing things by instinct or automatically. System is defined as a set of things or ideas working together as a whole [5]. It is a collection of objects and activities, plus a description of the relationships that tie the objects and activities together. Therefore, intelligent system refers to a set of activities that utilized the concepts and response rules to reach a final objective.

Intelligent system is fundamentally a stimulus-response system. The stimulus is the communications entering through the needs. The system extracts information from this and represents it as a situation. Then, the system selects a response rule, appropriate to the situation (which allow the system to get nearer to its objective), and performs the response part of this rule. The system makes its selection of response rules from those that it finds stored in its memory that was the accumulated

response rules that it has generated from earlier experiences and from generalizations based on previously elaborated response rules.

#### 1.1.2 Definitions for "Schedule"

There are two definitions for "schedule":

A schedule is a plan that gives a list of evens or tasks together with the times at which each thing should be done [5].

Or

A schedule or timetable is a placement of a set of meetings in time [10].

A meeting is a combination of resources such as rooms, people and items of equipment. Some of these resources may specify by the problem, and some must be allocated as part of the solution.

#### 1.1.3 Course Scheduling System

Course scheduling system is the problem of assigning times and places to a many separate courses to satisfy several constraints concerning capacities and locations of available rooms, free-time needs and other such considerations for lecturers and relationships between particular courses. The most prominent overall constraint (central to all timetabling problem) is that there should be no **clashes**, which means that any pair of courses which are expected to share common students or lecturers should not be scheduled simultaneously.

Course Scheduling or timetabling is a complex task. One wishes to schedule courses so that students can take the classes they want, without timing conflicts, so that faculty desires are accommodated, and so that facilities are effectively utilized. Constraints include room capacity and other attributes, classes that must be scheduled at the same time or on the same day as other classes, faculty that may be available only at limited times, and having a minimum "downtime" between classes.

4

### 1.2 The Course Scheduling Problems

Timetabling is a particular form of the general scheduling problem. There are a set of events and a set of processes that need to be carried out for each of those events. In an educational context, scheduling is a problem that must be concern for every teaching institution. Every semester a new schedule must be produced to take into account changes in the staffs, students and courses. This often involves a large amount of administrative work. Due to the difficulty in constructing a schedule, a considerable attention has been devoted to automated scheduling. During the last thirty years, starting in the early 60's with Gotlieb (1963) and others, many researches related to automated scheduling have been done and several applications have been developed and employed successfully.

#### 1.2.1 Course Scheduling at FCSIT

Several problems were faced while generating a course schedule for Faculty of Computer Science and Information Technology (FCSIT). The major problem is the actual size of the problem. The schedule for FCSIT covers more than 50 different courses, 40 lecturers and thousand over of students each semester and these numbers are believed to increase in the coming year.

The next problem is the variety and large number of constraints. These constraints sometimes are non-compatible. Due to the increasing number of courses, lecturers and students, the constraints on a particular schedule are getting more.

#### 1.2.2 Issues in Course Scheduling

A schedule generally has an aim, specific constraints and resources. These must be elicited before the process of constructing a schedule can start. Some of the example issues are givens in figure.

> What is the purpose of the schedule? What constraints must the schedule satisfy? Does a feasible schedule exist? Who will be affected by the schedule?

Figure 1.1 Issues in Schedule Construction.

Each of the issues mentioned are relevant to the course schedule and each one may be addresses individually. The purpose of the schedule, for example, is to ensure that all students take every course they are required to, while maintaining a reasonably efficient use of resources. It must satisfy certain constraints, in particular that no student is required to be in more than one course simultaneously.

## 1.2.3 Scheduling Terms

Scheduling problem has its own associated terminology and the terms associated with course scheduling problem are not specific. Below are the scheduling terms and their meanings.

Terms	Manit
Conflict/Clash	Two courses conflict if they are scheduled to the same
Constraint	A constraint places a restriction on when or where courses
Course	An event or meeting involving specific set of lecturers, students rooms and possible etc.
Feasibility	A schedule is feasible if it acti f
Period	A period is a fixed timeslot in which courses may be scheduled
Resource	Resources are physical entities that are referred to by the schedule. For example, lecturer, student, room and
Room	A venue where a
Session	All the times of t
Schedule	A description of the movement and grouping of resources over time.

Table 1.1 The list of scheduling terms and their meanings.

### **1.3 Objectives**

The main objective of this project is to automate the scheduling system given input parameters such as courses, characteristics of courses, constraint on rooms, time and lecturer availability. This project is very important since the number of courses offer, the lecturers and students are increasing and these make the schedule construction getting more and more complicated and it will be a difficult work to solve the problem manually. The automated scheduling system should:

- speeds up the generation of course schedule
- provides ease of gathering and manipulation of the lecturers, classes, rooms, courses and other records
- has the capabilities of reducing the time and effort for generating the schedule
- produced schedules that are feasible and with sufficient quality to be used
- be flexible enough to handle the variety of constraints encountered in real-life problems
- be user-friendly which means that easy to learn, easy to use and helpful
- be easy to maintain

Chapter 1 Introduction

### 1.4 Scope

This project typically not only covers on the faculties in University of Malaya but also can be used for the others faculties in all universities. However, the major concern here is to schedule courses that offer by Faculty of Computer Science and Information Technology (FCSIT) in University of Malaya and courses that will be held in this faculty.

Courses that offer by this faculty including the 1<sup>st</sup> year Degree, 2<sup>nd</sup> year Degree and 3<sup>rd</sup> year Degree courses in Computer Science and Information Technology. These courses can be a core subject, an elective subject or a lab course and sometime a course might be a combined course for two or more different year of students.

Chapter

2

Literature Review

# 2.1 Automated vs. Interactive Scheduling

Most of the people believe that the scheduling problem cannot be done full automatically by the system. There are two reasons why they think like that. The first reason is there are some reasons that make a schedule better than the others' schedules, which cannot easily, determined by an automatic system. The second reason is it is too complicated or too difficult for a system to search for a schedule that can satisfy all the constraints. A human intervention may be needed to produce an ideal schedule, which the system may be not able to find a solution itself. So, most of the scheduling systems enable user to adjust the final schedule manually. These systems are called the interactive or semiautomatic scheduling system.

The size and complexity of the scheduling system have provoked a trend towards a more general problem solving algorithms with the **automated** system. A full search of all possible schedules is not acceptable for the users who want to produce the schedule in a short time. The scheduler must able to solve this problem in return for a gain in speed. This could be done by an automated system.

### 2.2 Solution Approaches

Many different methods have been developed and used successfully to solve real scheduling problems. These methods are used in one particular department or institution only, and there is no benchmark to facilitate which approach is the best solution to scheduling problem.

Later on, researchers started to apply general techniques to the scheduling problem. The approaches taken included integer programming, network flow, graph coloring and others. But, recently, this problem has been tackled with techniques belonging also to artificial intelligence such as logic programming, simulated annealing, tabu search, genetic algorithms and constraints satisfaction.

### 2.2.1 Heuristics Algorithms

Heuristics is a term derived from a Greek word meaning to discover [14]. A heuristic is a rule of thumb, strategy using some knowledge about particular problem domain, which is used to guide search processes towards acceptable problem solutions. It guides to the lines that have high probability of success while avoiding wasted effort [12].

A variety of heuristic techniques have been devised to improve the efficiency of the search process. These fall into two categories: general-purpose heuristics and domain specific heuristics. General-purpose heuristics are flexible procedures that can be used on a wide variety of problems to prune the search space. Domain specific heuristics are special procedures that apply only to the given problem. One or both types of heuristics may be used depending upon the problem. In both cases,

the goal of the heuristic is to provide knowledge that eliminates the problem of combinatorial explosion and shortens the search time [14].

The scheduling methodology described in Figure 3.1. Algorithm 1 finds a nonconflicting set of courses and Algorithm 2 assigns these selected courses to rooms. The process is circulated for each period until all courses have been scheduled to a period and a room without conflicting. The unselected courses may be taken as a basis for the search of courses for the next period since they will have no conflicts between them, but also will not conflict with any exams from the previous period [7].



Figure 2.1 A heuristic scheduling framework.

One limitation in this approach is that heuristic assignment algorithms do not easily allow for the possibility of search and strongly limit the type of constraints that may be created for. Some of the constraints were viewed, as hard constraints (cannot be violated) whereas others may be viewed as soft (may be violated if necessary). It is possible to have a situation where the hard constraints may be ignored, which caused by the large number of soft constraints as the way they are represented [7].

### 2.2.2 Evolutionary Algorithms

Concurrent with part of the investigation, Lemoine developed a code for the pipeline-scheduling problem, which used evolutionary programming [1]. An evolutionary algorithm works on a set of possible solutions to a problem. Each solution is encoded as a gene (a string of symbol) [6].

Therefore, evolutionary scheduling typically begin with defining the chromosome representation of the event. Next the constraints are translated into a fitness function. Appropriate combination and mutation operators heave to be decided. Then, the selection, evaluation and breeding steps are carried out to determine the best solution for the problem [16].

### 2.2.3 Genetic Algorithms

The genetic algorithm begins with a schedule generated by the initial application of the greedy heuristic [1]. Genetic algorithms are powerful optimization tools that may be applied to a complex scheduling problem. They often capable to find an optimal solution even in the most complex of search space [8].

The algorithm uses two basic operations, mutation and crossover. Given a schedule, a mutation replaces a random number of randomly selected classes with randomly

selected classes of the corresponding course and shift. A crossover is slightly more complicated and involves combining two schedules [1].

### 2.2.4 Graph Coloring

In this approach, each event is associated with a vertex in a graph, and there is an edge between each pair of event that cannot be scheduled at the same time. For example, lectures hat share a common lecturer or common students are joined. Unavailability and preassignments are managed imposing some external constraints on the colorability of specific vertices of the graph [3]. Each vertex is then colored such that no pair of vertices connected by an edge has the same color [8].

A coloration of the resulting graph, respecting the external constraints, can be easily turned into a schedule, by assigning a period to each color, and consequently scheduling the events corresponding to a vertex to the period corresponding to its color [3].

### 2.2.5 Memetic Algorithms

Memetic algorithms are an extention of genetic algorithms, based on a model of how an idea evolved. A **meme** is the basic unit of information that reproduces itself as a result from people exchange idea. A meme differs from gene as it can be improved during their lifetime. Genes usually were passed between individuals unaltered but individuals adapt the meme if it is better [9,10].

The advantage gained with using the Memetic algorithms is that the space of possible solutions is reduced to the subspace of local optima.

Each solution in the population is represented as a number of memes and each of these contains information. Initial population is generated using weighted wheel. Then this population is crossover using a combination of light and heavy random mutation, followed directly by application of hill climbing techniques. After that, evaluation function is applied to it to calculate the fitness. In order to reduce the population, individuals are chosen from the pool with a section function to joint the new population. This is achieving by using the roulette wheel selection to weight the fitness of an individual to its competition [9].

# 2.2.6 Network Flow Techniques

In this technique, network model was used as the core of the scheduling algorithm. The scheduling problem was reduced to a sequence of network flow problems. A network is created for each event so that the flow in the network identifies constraint and resources given. The construction of the network is repeated for all events eventually. If a solution is found for all the networks, it leads to a complete schedule [3].

But there is one limitation in this approach that there is no backtracking on the events already scheduled, as a result there is no guarantee that the solution is found whenever it exists. Therefore the procedure solves the scheduling problem if it finds a feasible solution, otherwise a human intervention changes some of the constraints manually so the reason of unfeasibility is get rid [3].

#### 2.2.7 Simulated Annealing

Simulated annealing is a probabilistic local search technique for findings optimize solutions to scheduling problem [3]. Simulated annealing is a search strategy, which keep track of one feasible schedule [14].

This technique starts by creating a random initial solution. The main procedure consists of a loop that generates a neighbor of the current solution, slightly altered at random from the current one. This neighbor is accepted as the current schedule if it has a lower penalty. If the neighbor has a higher penalty, it may be accepted according to a probability that related to a control parameter called temperature. The temperature of inferior neighbors is decreasing after a particular number loops. This procedure stop when a solution close to objective is found and no solution that increases the objective function is accepted any more, which the system is frozen.

One limitation with simulated annealing is that the process can take a long time to stop in order to achieve a good solution.

#### 2.2.8 Tabu Search

Tabu Search is a local search technique, family of general-purpose techniques for the solution of optimization problem [3]. The local search techniques start from an initial solution, enter an iteration that navigate the search space, stepping iteratively from one solution to its neighbors [4].

Tabu Search remembers just one current feasible schedule. A Tabu Search maintains a list of tabu moves to prevent recycling. A tabu list represent the schedule which having visited recently are forbidden in order to prevent the search staying in the same area. This tabu list is usually with a fixed size, with first-in-first-out basis. An aspiration level, which represents the best solution visited, is maintained. If a tabu schedule reaches at the aspiration level, it may be removed from the tabu list [14].

The Tabu Search stops either when the number of iterations reaches a given value or when the value of the objective function in the current solution reaches a given lower bound.

#### 2.2.9 Constraint Logic Programming

Constraint logic programming (CLP) system is a tool for modeling a specific search problem, which provides the ability to declare variables and their domains, and constraint on the problem [3]. CLP generalizes logic programming by replacing unification with constraint solving over a particular domain with a set of discrete variables [11].

In CLP, a labeling strategy dictates the order in which the search space is traversed. There are two orders, which should be take note: the order the variables are instantiated and the order in which the variables' values are assigned [14].

In order t search for a solution, a CLP system generates values for the variables, propagating values through the constraints in order to prune parts of the solution space where inconsistencies are discovered. Therefore, this method is a backtrack search where the constraints allow the system to look ahead to the consequences of decisions and spot failures earlier [6].

CLP is a paradigm coined for solving combinatorial problems. It has been used successfully for a variety of practical applications from scheduling to financial analysis. Constraint handling has been introduced into Logic Programming to simplify the expression of problems and to dramatically improve program efficiency. Essentially the constraints are used to prune the search tree defined by the logic program in which they are embedded [2].

#### 2.3 Case Studies

This section provides some examples of current GA based timetabling systems that are actually being used to provide practical solutions in institutes, giving a flavor of what is currently achievable.

A commercially available evolutionary timetabling systems is ACT (Automated Class Timetabler) which is a state of the art system for university timetabling, running under Windows-95, developed in Korea. The system uses a hybridized algorithm, which incorporates hill climbing, a genetic algorithm and also a best-first heuristic search algorithm. It is able to solve all hard constraints and optimizes soft constraints in a two-phase approach. The system produces class timetables, professor timetables and room timetables. It has been successfully tested in 15 universities in Korea, and can cope with considerably large problems. In the largest university tested, timetables were evolved for 70 departments, with 700 professors, and 4000 subjects, satisfying 5 hard constraints and optimizing up-to 10 soft constraints. Using a Pentium 133Mhx processor, timetables can be produced in around 20 minutes, and hence the system is extremely fast. It contains a straightforward graphical user interface based on MS-Windows for data entry. An added feature is manual scheduling function that enables any lesson attributes to be changed after the automated procedure has finished, with guidance from the system as to the possible changes that can be made.

Several other universities are using implementations of evolutionary timetabling systems to schedule lectures and exams within the university. For example, at Napier University, in Edinburgh, a memetic algorithm based timetabling system has

#### Chapter 2 Literature Review

been developed, known as Neeps and Tatties. The approach and representation used in this case is entirely different from that described above. The university to schedule the entire university timetable has successfully used Neeps and Tatties. This is an extremely large problem --- for example, in the first semester of 1997 there were 2067 events to place in 45 time-slots and 183 rooms. These were attended by 669 lecturers and 978 student groups. The timetables produced had to optimize 12 different criteria --- the algorithm was run on a single machine, and takes about an hour to satisfy all hard constraints. From then on, it begins to optimize the soft constraints and can be run indefinitely. In practice, the algorithm was left running for about three days. However, a great advantage of the program is that it can be seeded with previous results, which allows solutions to evolve gradually as the data is gradually entered, or that the program can be seeded with a solution from a previous year to speed things up. A novel feature of this program is the ability to interactively change the weights of the soft constraints as the program runs, and see the results with immediate effect, which is of invaluable help to an administrator trying to find a satisfactory timetable.

Another system was developed at the University of Edinburgh, where it is used each year to schedule examinations. This is known as GATT, standing for Genetic Algorithm Time Tabler. This software was also used by Harvard Business School to schedule their examinations, and was found to outperform a commercially available timetabling system in trials. In a direct head-to-head competition with the commercial (non-GA-based) software, exams were scheduled over 15 slots -- GATT found a schedule with 2 students with direct conflicts, involving 2 courses. The commercial software came up with 14 conflicts when running automatically,

#### Chapter 2 Literature Review

while their manual attempt to develop a schedule came up with fewer, with only 5 students with conflicts! GATT has also been tested on a real timetabling problem from a secondary school in Belgium, which involved scheduling classes and teachers for the first two-year groups. The problem involved 10 different class groups, 26 different teachers, and 318 different courses. In total, 5320 edge constraints were involved. The constraints included a significant number of specifications and exclusions, where an event was either pre-assigned to a particular slot, or excluded from a range of slots. Defever reports that GATT removed all clash violations. He does not specify however if other constraints, such as room constraints or proximity constraints were considered.

### 2.4 Programming Language

The programming language that used in this project is Microsoft Visual Basic 6.0.

#### 2.4.1 What is Visual Basic 6.0?

Microsoft Visual Basic 6.0 was mainly used in development and designing of Automated Scheduling System. Although the syntax of the Visual Basic language parallels that of previous versions of BASIC, Visual Basic represents a conceptually different approach to programming, called *event-driven programming*. An application developed with an event-driven model responds to events that happen in the computer environment. Such events include the press of a mouse button or a call from another application running concurrently.

#### 2.4.2 Why Visual Basic 6.0?

Visual Basic 6.0 is a user-friendly programming language. We can create the user interfaces for the applications by directly manipulating on-screen objects such as control buttons and dialog boxes. We assign value for the properties of these objects by selecting various characteristics such as colors and fonts from an extensive list. One of the great triumphs of Visual Basic is that we can develop complete Windows applications with minimal program instructions.

### 2.4.3 Features of the Software

The software used in this project is the Visual Basic version 6.0.

#### 2.4.3.1 Safe and Fast

Visual Basic meets the professional's demands for speed, performance and safety. It produces compact, fast and highly optimized native machine code, comparable to that of a C compiler.

#### 2.4.3.2 A Wealth of Facilities

All the components that make up the integrated environment can be used in the programs. A wide range of standard predicates provides all the features that expected from a professional programming language.

Chapter

3


### 3.1 Constraints

A constraint is a restriction on resources related to an event [16]. Scheduling constraints are many and varied. Constraints are usually divided into 'Hard' and 'Soft' constraints:

- Hard constraints. Hard constraints are a restriction that a schedule must follow.
   A schedule, which breaks this constraint, is not a feasible solution and must be reconstructed or rejected.
- Soft constraints. Soft constraints are less important compared to hard constraints. Usually it is impossible to avoid breaking at least some of these constraints [10].

# 3.1.1 Constraints on Course Scheduling

#### 3.1.1.1 Lecturer

No lecturer is schedule to two different courses at the same time.

#### 3.1.1.2 Course

No two same level and same major courses run simultaneously.

#### 3.1.1.3 Time

- Sessions for Monday to Friday are to hourly intervals from 8.00 a.m. to 6.00 p.m.
- There should not have any classes on Saturday and Sunday.

#### 3.1.1.4 Room

- There must not be more students scheduled to a room that more than the capacity of the room.
- There should not have two classes scheduled in the same time at the same room.
- Lab classes should be schedule to computer lab only.
- Tutorial classes must be scheduled in the tutorial room only.
- Courses with more students must be scheduled in the large rooms.

### 3.2 Rule-Based System

Rule-based system is defined as a computer program that processes problem specific information contained in the working memory with a set of rules contained in the knowledge base, using an inference engine to infer new information [13].

The rule-based expert system consists of a number of rules or heuristics and conventional recursion to assist in carrying out class assignments. In a rule-based system, the rules are contained in the knowledge base and also the working memory. These rules are combined through the inference engine to infer new information as shown in Figure 3.1.



Figure 3.1 Rule-based model.

#### 3.2.1 Forward Chaining

Forward-chaining is defined as "inference strategy that begins with a set of known facts, derives new facts using rules whose premises match the known facts and continues this process until a goal state is reached or until no further rules have premises that match the known or derived facts" [13].



Figure 3.2 Forward-chaining inference process.

Figure 3.2 shows the forward-chaining inference process. First, the system will insert information into the working memory. Then the inference engine checks the rules in the working memory and it will fire new rules to the working memory from its rules' conclusion. The facts in the working memory are continually updated. Rules in the system represent possible actions to take when specified conditions hold on items in the working memory. The conditions are usually patterns that must match items in the working memory, while the actions usually involve adding or deleting items from the working memory.

#### 3.2.1.1 Advantages of Forward Chaining

Forward chaining is a good approach to the problem, which begins with gathering information, and then new rules and facts are inferred from this information. Therefore, this method can provide a large amount of facts even though the data provided is in small amount only [13].

#### 3.2.1.2 Disadvantages of Forward Chaining

This method does not recognize which facts or rules are more important than the others. Therefore, it sometimes took a longer time to come out with a solution. The system may produce unrelated questions or asking the questions in a wrong sequence to the user [13].

#### 3.2.2 Backward Chaining

Backward chaining is defined as "inference strategy that attempts to prove a hypothesis by gathering supporting information" [13].

Backward Chaining often referred to as hypothetical reasoning starts with a specific hypothesis, or set of hypotheses, called the agenda. The agenda structures knowledge and controls backward-chaining event processing by ordering hypotheses or goals, in a numbered, hierarchical outline. The backward-chaining inference engine works backward from the agenda, pursuing a hypothesis via its search order strategies.



Figure 3.3 Backward-chaining process.

Figure 3.3 shows the backward-chaining process. The backward-chaining inference engine start with the first goal in the knowledge base's agenda. The goal will be the attribute of the domain class or an instance.

Then inference engine backtracks through the goal's search order. This search order provides the backward-chaining inference engine where, and the order to obtain the value needed to prove the goal. Combination of the session context, a when needed method, the knowledge base's rules, and end-user query, or the default values were used to obtain the value.

If the goal can be verify through rules, the inference engine backtracks through the goal's rule group (set of rules that can conclude the goal). If an attribute in the

premises of a rule can conclude the goal, then the inference engine backtracks trough that fact search order to find its value. This process repeated until the final goal is reached.

### 3.2.2.1 Advantages of Backward Chaining

This approach is a good technique to follow if the problem begins with validates the hypothesis. The chaining system focused on a given goal only. Therefore, the question ask to the users are related. Backward chaining only search for the rules that are related to the goal only [13].

### 3.2.2.2 Disadvantages of Backward Chaining

The system will continue to follow a line given for reasoning although that particular hypothesis cannot be proved. As a result the system did not stop [13].

### 3.3 Designing Backward Chaining System

# 3.3.1 Combination of Heuristics Algorithms and Backward Chaining

A combination of heuristic algorithm and backward-chaining approach will be used to solve the scheduling problem in this project.

Using heuristics will guide to search processes towards acceptable problem solutions. From the expert knowledge and the natural way of doing, it will be easy to produce feasible schedule by following some rules. In the scheduling case, the hard constraints should be given a priority over the soft constraints. In this project there are few heuristic algorithms as below:

- Find a core course, which is a combined course with a preferred timeslot and fixed this course into the schedule.
- Find a core course, which is not a combined course with a preferred timeslot and fixed this course into the schedule.
- Find an elective course, with a preferred timeslot and fixed this course into the schedule.
- Find a core course, which is a combined course and fixed this course into the schedule.
- Find a core course, which is not a combined course and fixed this course into the schedule.
- 6. Find an elective course and fixed this course into the schedule.

Algorithm 1 finds a core course, which is a combined course with a preferred timeslot and fixed this course into the schedule. This algorithm is repeated until all

the core courses, which is a combined course with a preferred timeslot have been scheduled to a period and a room without conflict. Then the system move to Algorithm 2 to finds a core course, which is not a combined course with a preferred timeslot and fixed this course into the schedule. The process is then continues same as previous algorithm.

The backward-chaining methodology approach had been chosen as the basic in this project. This is because the scheduling process always starts with hypothesizing a solution where to fix a time for a particular event and then check whether this is valid without violating any other constraints. At the same time, more data are needed to make conclusions in the scheduling process, which the backward-chaining approach is more suitable here. In addition, the backward-chaining system search for the rules that are relevant to the goal only compare to forward-chaining method that does not recognizes which rules are more important. Therefore, backward chaining will be the more suitable approach to solve the scheduling problem compare to forward-chaining approach that sometimes took a longer time to come out with a solution.

#### **3.3.2 Defining the Problems**

The scheduling problem is study detailed here. The major task in this project is to develop an automated scheduling system to assist a scheduler with the production of new schedule.

Information relates to scheduling problems such as courses offer, details about a course, information on lecturers are being collected. Details on a course such as the

35

code, course's name and credit hours as well as lecturer information such as lecturer code, name and courses taken by the lecturer each new semester are collected too.

#### 3.3.3 Defining the Goals

Coding of a backward-chaining system must begin with defining the system's goal. The automated scheduling system will have two principal goals to achieve:

- Determine the time-slot for a course.
- Determine the venue for a course.

A time-slot is the period where the course can be place on the schedule. The timeslot or period for a particular course must be in between the session determined by the user. The structure of the time-slot for a course is determined from user input to the system too. To determine whether a particular time-slot is available for a course, two situations that must be concern:

- Students availability
- Lecturer availability

There are 3 different goals that this system to pursue. For a course and a particular time,

- All students involved available at the particular time.
- The lecturer involved available at the particular time.
- The particular room available at the time.

A venue for a course is a room where the course will be held in that particular time. To determine whether a venue is available at the time, situation that must be concern:

Room availability

#### 3.3.4 Defining the Goal Rules

In this project, the system has to fix the course into the schedule, which is the rules' conclusion. Therefore, necessary preconditions to satisfy the rules' conclusion should be determined. The general form for all goal rules will look like following

[13]: IF Precondition\_1

AND Precondition 2

#### **THEN Conclusion**

The principal issues considered when making a rule's conclusion in this system:

Student availability

THEN room av

- Type of course (core subject or elective subject)
- Course structure (ordinary course or combined course)
- Lecturer availability
- Other courses at the same time (yes or no)
- Room availability
- Room size (more than or less than students' size)
- Equipment
- Special room (lab class or not a lab class)

Chapter 3 Methodology

#### 3.3.5 Expanding the System

Then the system is expanded to cover a broader understanding of the problem. The rules are sub divided according to the type or course whether is a lab course or not.

3.3.5.1 Course Rules

IF room available AND lecturer available AND student available THEN time available for course

IF student's size less than room's size AND no other course at the same time in the room THEN room available for course

IF lecturer not having other course at the same time THEN lecturer available for course

IF student not having other conflict's course THEN student available for course Chapter 3 Methodology

Automated Scheduling System for FCSIT

### 3.3.5.2 Lab Course Rules

IF lab available

AND student available

THEN time available for lab course

IF student's size less than lab's size AND no other course at the same time in the lab THEN lab available for lab course

IF student not having other conflict's course THEN student available for lab course

# 3.4 Algorithms for Course Scheduling System

First, the system will find a timeslot for a subject. If the timeslot value is consistent, then an available room that satisfies constraints is selected for that particular timeslot. If the room is not available, then the next timeslot is selected by the random function. Consistency tests will be performed on all new values selected by the random function. The purpose of consistency test is to determine whether values from the domains of related variables are consistent (non-conflicting) or not with respect to constraints. If not consistent, the values will be removed from its domains [17]. The algorithm is showed in Figure 3.4.



Figure 3.4 The algorithm for course scheduling system.

# 3.5 Processes and Flows in the System

Data flow diagram is a graphical representation or data process through out the system. These diagrams provide an overview of system inputs, processes and outputs, which correspond to the general systems in the project.

The following section shows the data flow diagrams in this project. The symbols used were explained in the table below.

Symbols	Meaning
	Entity – Any object or event which data is collected. Entity may be a person, place or thing.
>	Flow of data – It shows movement of data with head of the arrow pointing toward the data's destination.
	Process – Process denotes a change or transformation of data. It is the work being performed by the system.
	Data store – This represents a data store and it may represent a manual store or a computerized file.

Table 3.1 The basic symbols used in the data flow diagram.



Figure 3.5 Context level diagram for the automated scheduling system.



Figure 3.6 Diagram of the automated scheduling meteos.



Figure 3.6 Diagram of the automated scheduling system.



Figure 3.7 Diagram of process on the course description.



Figure 3.8 Diagram of process on the room description.



Figure 3.9 Diagram of process on the lecturer description.



Figure 3.10 Diagram of process on the session description.

Chapter

4

# System Design

### 4.1 Automated Scheduling System Design

This scheduling system design is based on the rule-based system. The system users should be able to generate a schedule from the data provided to the system. These data include course information, lecturer information, room information and schedulable time information as shown in Figure 4.1.



Figure 4.1 Major system design for automated scheduling system.



Figure 4.2 Sub-modules in course entries modules.



Figure 4.3 Sub-modules in room entries modules.



Figure 4.4 Sub-modules in lecturer entries modules.



Figure 4.5 Sub-modules in schedulable time modules.

#### 4.2 Database Design

All the data involved in this system, created by the user of the system were stored in the database files (.mdb) in the Microsoft Access form.

### 4.3 User Interface Design

User interfaces (UI) can be tricky things to design, because different users have different preference in the style of perceiving, understand, and working. The UI for this scheduling system are not very complex because the Microsoft Visual Basic 6.0 does not provide a lot of features for the interface design. The main focus in the interface design is to enable users easily input all the necessary data.

## 4.3.1 Course Information Entries Window Design

This window is to let users input the information for a course including the course's code, course's name, credit hours, number of students taking the course, lecturers attached, discipline and level of the course. In addition, questions such as whether the course is a lab course, a core or elective course, a combined course also needed by the system.

Course Code :	(A)
Course Name :	
Credit Hours : 3	A REAL
Students Size:	Course Categories
Lecturers Attached : 1.	© Yes C No
2.	Course Type: Fearthy
Students Categories	recury
Discipline : Computer Science	Combined Course Yes
Level: First Year	Combined with

Figure 4.6 Course information entries window.

## 4.3.2 Room Information Entries Window Design

This window is to let users input the information on the rooms available in scheduling the course. Information such as room's code, name, capacity, and type of room is needed for the rooms.

AUTOMATED SCHEDULING SYSTEM - [Room Information Entries]			
System Input View Schedule About			
		and the second second	
			and the second
Room Code :			
			A Merry
Room Name :			
Size Of Room :			State State
			Equal Section 1
Room Type			
G Lecture Hall C Transfel Orac			The second second
			An and an area
C Lecture Room C Computer Lab			
C Auditonum			
	and the second of		a market and a
Save		AND A STREET	
Clack	Ca	ncel	A State State
	8/24	1/01	4:24 PM

Figure 4.7 Room information entries window.

# 4.3.3 Lecturer Information Entries Window Design

This window allows users to key in the lecturers' information such as their codes and names. Then the courses lectured by the lecturers in that semester must be specifying if there is any.

A AUTOMATED SCHEDULING SYSTEM - [Lecture: Information Entries] S. System Input View Schedule About E. S. B. E. E. E. E. E.	
Lecturer Code :	
Lecturer Name :	- at
Courses Attached : 1.	
2.	
Save	Cancel
	8/24/01 4/24 PM

Figure 4.8 Lecturer information entries window.

# 4.3.4 View Courses Information Window Design

Users can view the available courses by enter data in this form including the discipline and the level of the courses.

View Bu			C	ourse Legend	
Tiew by		Generate	List	CS First Year	
Discipline		Low sector and the sector of t		CS Second Year	
Level:		Clear List		CS Third Year	
		Cancal		IT First Year	
	The second s	Care		IT Second Year	
	And the second second second second			IT Third Year	
elected Cour	SBB LIST				
Course Code	Course Name	Credit Hours	Size of students	Lecturers Attached	
elected Cour Course Code WXES1101	Course Name Pengenalan Kepada Pengaturcaraa	Credit Hours 3	Size of students	Lecturers Attached	
elected Cour Course Code MXES1101 MXES1109	Course Name Pengenalan Kepada Pengaturcaraa Sistem Dan Organisasi Komputer	Credit Hours 3 3	Size of students 300 300	Lecturers Attached Norisma,RJRY Bukaini,WCS	
elected Cour Course Code WXES1101 WXES1109 WXES1301	Course Name Pengenalan Kepada Pengaturcaraa Sistem Dan Organisasi Komputer Sistem Pengendalian	Credit Hours 3 3 3	Size of students 300 300 300	Lecturers Attached Norisma,RJRY Rukaini,WCS ZR.RM	
Directed Cour Dourse Code MXES1101 MXES1109 MXES1301 MXES1401	Course Name Pengenalan Kepada Pengaturcaraa Sistem Dan Organisasi Komputer Sistem Pengendalian Pengenalan Kepada Sistem Komput	Credit Hours 3 3 3 3 3	Size of students 300 300 300 250	Lecturers Attached Norisma,RJRY Rukaini,WCS ZR,RM MKO	
elected Cour Course Code WXES1101 WXES1109 WXES1301 WXES1401 WXES2103	Course Name Pengenalan Kepada Pengaturcaraa Sistem Dan Organisasi Komputer Sistem Pengendalian Pengenalan Kepada Sistem Komput Pangkalan Data	Credit Hours 3 3 3 3 3 3 3 3 3	Size of students 300 300 250 300	Lecturers Attached Norisma, RJRY Rukaini, WCS ZR, RM MKO Nur, Azlina, Bafidab	
Elected Cour Course Code MXES1101 MXES1109 MXES1401 MXES1401 MXES2103 MXES2201	Course Name Pengenalan Kepada Pengaturcaraa Sistem Dan Organisasi Komputer Sistem Pengendalian Pengenalan Kepada Sistem Komput Pangkalan Data Pengaturcaraan Berorientasikan Ob	Credit Hours 3 3 3 3 3 3 3 3 3 3 3 3 3	Size of students 300 300 250 300 300 300	Lecturers Attached Norisma, RJRY Rukaini, WCS ZR, RM MKO Nur Azlina, Rafidah LSP, Bafidah	
Altected Cour           Course Code           WXES1101           WXES1109           WXES1301           WXES1401           WXES2103           WXES2201           WXES2302	Course Name Pengenalan Kepada Pengaturcaraa Sistem Dan Organisasi Komputer Sistem Pengendalian Pengenalan Kepada Sistem Komput Pangkalan Data Pengaturcaraan Berorientasikan Ob Komunikasi Data Dan Rangkain Kon	Credit Hours	Size of students 300 300 250 300 300 300 300	Lecturers Attached Norisma,RJRY Rukaini,WCS ZR,RM MKO Nur Azlina,Rafidah LSP,Rafidah 07 IB	
Altected Cour           Course Code           WXES1101           WXES1109           WXES1301           WXES1401           WXES2103           WXES2201           WXES2302           WXES2401	Course Name Pengenalan Kepada Pengaturcaraa Sistem Dan Organisasi Komputer Sistem Pengendalian Pengenalan Kepada Sistem Komput Pangkalan Data Pengaturcaraan Berorientasikan Ob Komunikasi Data Dan Rangkain Kon Senibina Sistem Komputer	Credit Hours 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3	Size of students 300 300 250 300 300 300 300 200	Lecturers Attached Norisma, RJRY Rukaini, WCS ZR, RM MKO Nur Azlina, Rafidah LSP, Rafidah OZ, IB NMN	
Altected Cour           Course Code           WXES1101           WXES1109           WXES1301           WXES1401           WXES2103           WXES2201           WXES2302           WXES2401           WXES2401           WXES2470	Course Name Pengenalan Kepada Pengaturcaraa Sistem Dan Organisasi Komputer Sistem Pengendalian Pengenalan Kepada Sistem Komput Pangkalan Data Pengaturcaraan Berorientasikan Ob Komunikasi Data Dan Rangkain Kon Senibina Sistem Komputer Amalan Pengaturcaraan Berorientas	Credit Hours 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3	Size of students 300 300 250 300 300 300 200 300 300	Lecturers Attached Norisma, RJRY Rukaini, WCS ZR, RM MKO Nur Azlina, Rafidah LSP, Rafidah OZ, IB NMN LSP, Rafidab	
elected Cour Course Code #XES1101 #XES1109 #XES1401 #XES2103 #XES2201 #XES2201 #XES2302 #XES2401 #XES2470 #XES2470	Course Name Pengenalan Kepada Pengaturcaraa Sistem Dan Organisasi Komputer Sistem Pengendalian Pengenalan Kepada Sistem Komput Pangkalan Data Pengaturcaraan Berorientasikan Ob Komunikasi Data Dan Rangkain Kon Senibina Sistem Komputer Amalan Pengaturcaraan Berorientas Pengenalan Kepada Pengeturcaraan	Credit Hours 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3	Size of students 300 300 250 300 300 300 200 300 300 300 300	Lecturers Attached Norisma,RJRY Rukaini,WCS ZR,RM MKO Nur Azlina,Rafidah LSP,Rafidah OZ,IB NMN LSP,Rafidah DS CTY	
elected Cour Course Code WXES1101 WXES1401 WXES1401 WXES2103 WXES2201 WXES2201 WXES2401 WXES2470 WXET1101 WXET1109	Course Name Pengenalan Kepada Pengaturcaraa Sistem Dan Organisasi Komputer Sistem Pengendalian Pengenalan Kepada Sistem Komput Pangkalan Data Pengaturcaraan Berorientasikan Ob Komunikasi Data Dan Rangkain Kon Senibina Sistem Komputer Amalan Pengaturcaraan Berorientas Pengenalan Kepada Pengaturcaraa Sistem Dan Organisasi Komputer	Credit Hours 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3	Size of students 300 300 250 300 300 300 300 200 300 300 300 300	Lecturers Attached Norisma,RJRY Rukaini,WCS ZR,RM MKO Nur Azlina,Rafidah LSP,Rafidah OZ,IB NMN LSP,Rafidah PS,CTK	

Figure 4.9 View courses information window.

# 4.3.5 View Rooms Information Window Design

Users can view the available rooms for the uses of lectures or computer labs on that semester.

Room Code	Room Name	Size of Doom	Danse Trees	Room Legend
AUDI	Auditorium	100	resona type	THA Lochico Hall
BK1	Blilk Kullah 1	00	AUDI	
BK2	Billik Kullah 2	60	LR	LR: Lecture Room
вка	Billk Kullah 3	60	LR	AUDI : Auditorium
BK4	Billk Kuliah 4	60	LR	TR: Tutorial Room
BK5	Billik Kuliah 5	63	LR	TAR. Commuter Lab
OK1	Dewan Kullah 1	300	LR	man, computer Lau
DK2	Dewan Kuliah 2	300	LI	
MM1	Makmal Mikro 1	50		
MM2	Makmal Mikro 2	50	LAB	
MM3	Makmal Boole	60	LAB	
MM4	Makmal Von Neuman	60	LAB	and a global participation of the
MM5	Makmal Rangkaian	60	LAB	and the second second second
AM6	Makmal Dijikstra	60	LAB	
		Ó		

Figure 4.10 View rooms information window.

# 4.3.6 View Lecturers Information Window Design

Users can view the available lecturers with the courses they are taken in that semester.

Lecturers List			
Lecturer Code	Lecturer Nome	Courses Attached	Generate List
AA	Abrizah Abdullah	WMET3102	
СТК	En Chiew Thiam Kian	WXET1101	and the second second
FH	Pn Farizah Hanum Md Nasaruddin	WXET2103	Clear List
Fizik	Jabatan Fizik	SMES2403	
HZ	Pn Hannyzzura Pal @ Affal	WRES3401,WXET1110	-
18	En Ibrahim Abu Bakar	WRES3303,WXES2302	Cancel
Kiran Kaur	Kiran Kaur	WMET3106	and the second second second
Law	Faculty of Law	LXEB1303	
LSP	Prof Madya Dr Lee Sai Peck	WKES3202,WXES2201,WXES247	
Maizatul	Maizatul	WXET2103,WXET3204	
Mati	Jabatan Matematik	SJEW1102,SJEW2301	
Mat2	Jabatan Matematik	SJEW1102	Sandar Print St.
Media	Jabatan Media	AKEA2124,AKEA2316,AKEA3308	
MKO	En Mohd Khalit Othman	WXES1401	and the second second second second second
ML	Ph Miss Leiha	WXET2470,WXET2305,WXET110	
NA	Cik Nor Aniza Abdullah	WRET3103	
NEN	Nor Edzan Haji Nasir	WMET2109	
		WRET2102	and the second second
	En Noorzally Mond Noor	WXES2401,WXET1109	
	Pri Nonzan Mohd Yasin	WMES3201,WMES3304	
Nonsma	Nonsma	WAES2201.WXES1101	Maria and a maria a log and and

Figure 4.11 View lecturers information window.

# 4.3.7 View Students' Schedule Window Design

Users can view the students' schedule filtered by students' discipline and level in that semester.

	IT First Year	rse Legend	Cou		A CONTRACTOR OF THE OWNER
24	IT First Year	The Loyeilu	The set of		-View By
tar	AT LOST I COL	LS First Year		All	Discipline :
Carl Carl Carl	IT Course little	CS Second Very		and the second s	1 Martin California
	II Second Year	Co Securio Teat		All	Level:
	IT Third Year	LS Insid Year			
	7R RM	DK1	5.00pm - 6.00pm	Tuesday	WXES1301
	ZR.RM	DK1	1.00pm - 5.00pm	Tuesday	AXES1301
· make a	ZR RM	DK1	8.00am - 9.00am	Wednesday	WXES1301
Defide	Nur Arline D	DK2	3.00pm - 4.00pm	Monday	AXES2103
Refide	Nur Azline B	DK2	4.00pm - 5.00pm	Monday	AXES2103
Bafida	Nur Azlina B	DK2	5.00pm - 6.00pm	Monday	WXES2103
ALT CLASTICE CA THE	LSP Befidah	DK2	8.00am - 9.00am	Tuesday	WXES2201
tab			and the second second is the second s	anger .	LA/E 09901
All a.F	ZR RM ZR RM ZR RM Nur Azline, F	Rooms Attached DK1 DK1 DK1 DK2 DK2	1.00pm - 5.00pm 5.00pm - 6.00pm 8.00am - 9.00am 3.00pm - 4.00pm 4.00pm - 5.00pm	Tuesday Tuesday Wednesday Monday Monday	XES1301 XES1301 XES1301 XES1301 XES2103 XES2103

Figure 4.12 View students' schedule window.

# 4.3.8 View Lecturers' Schedule Window Design

Users can view the lecturers' schedule filtered lecturer code in that semester.

In the second	chedule	Lecturers Schedul	P	tooms Schedule	
View By		Cou	beene lean		
			CS Fast Year	I me av	
Lecturer Code	a: LSP		COTTANTON	II Fist Year	
	Children of the second s		LS Second Year	IT Second Year	
			CS Third Year	IT Third Year	
1 1 1 1 1	Mandau	9.00.000 8.00.00	WKEG3202	DKI	
	Monday	9.00 cm 9.00 cm	WKEG3303	FLOOM ALLECTED	
LSP	Monuay	0.00am - 3.00am	77 DL-03-12 UZ		1012/085
LSP	Monday	9.00am - 10.00am	WKES3202	DK1	题
LSP LSP LSP	Monday Monday	9.00am - 10.00am 10.00am - 11.00am	WKES3202 WKES3202	DK1 DK1	No.
LSP LSP LSP LSP	Monday Monday Tuesday	9.00am - 10.00am 10.00am - 11.00am 8.00am - 9.00am	WKES3202 WKES3202 WKES3202 WXES2201	DK1 DK1 DK1 DK2	
LSP LSP LSP LSP LSP	Monday Monday Tuesday Tuesday	9.00am - 10.00am 10.00am - 11.00am 8.00am - 9.00am 9.00am - 10.00am	WKES3202 WKES3202 WXES2201 WXES2201	DK1 DK1 DK2 DK2	
LSP LSP LSP LSP LSP	Monday Monday Tuesday Tuesday Monday	9.00am - 10.00am 10.00am - 10.00am 10.00am - 11.00am 8.00am - 9.00am 9.00am - 10.00am 11.00am - 12.00pm	WKES3202 WKES3202 WXES3202 WXES2201 WXES2201 WXES2201	DK1 DK1 DK2 DK2 MM1	
LSP LSP LSP LSP LSP LSP LSP	Monday Monday Tuesday Tuesday Monday Monday	9.00am - 10.00am 9.00am - 10.00am 10.00am - 11.00am 8.00am - 9.00am 9.00am - 10.00am 11.00am - 12.00pm 11.00am - 12.00pm	WKES3202 WKES3202 WXES3202 WXES2201 WXES2201 WXES2201 WXES2201	DK1 DK1 DK2 DK2 MM1 MM2	
LSP LSP LSP LSP LSP LSP LSP LSP	Monday Monday Tuesday Tuesday Monday Monday Monday	9.00am - 10.00am 9.00am - 10.00am 10.00am - 11.00am 8.00am - 9.00am 9.00am - 10.00am 11.00am - 12.00pm 11.00am - 12.00pm	WKES3202 WKES3202 WXES2201 WXES2201 WXES2201 WXES2201 WXES2201	DK1 DK1 DK2 DK2 MM1 MM2 MM3	A second s      Expecting second sec

Figure 4.13 View lecturers' schedule window.

# 4.3.9 View Rooms' Schedule Window Design

Users can view the rooms' schedule filtered by room code in that semester.

International Advantage of the Advantage o	SCREDUE	Lecturers Schedul	ler F	tooms Schedule
View By		Cou	Irse Legend	
			CS First Year	TT Frat Van
Room Code	DK1	-	CC Canand View	
	State State State		us second real	11 Second Year
			CS Third Year	IT Third Year
JKI	Monday	3.00pm - 4.00pm	SJEW1102	Mat1,Mat2
DKI	Monday	4.00pm - 5.00pm	SJEW1102	Mat1,Mat2
DK1	Monday	5.00pm - 6.00pm	SJEW1102	Mat1,Mat2
second state and a second s	the second se	8.00am - 9.00am	WXES1101	Noriema D IDV
DK1	Tuesday	or or an in or or call	TTT ILL GITTGI	rionand, rorti
DK1 DK1	Tuesday	9.00am - 10.00am	WXES1101	Norisma,RJRY
DK1 DK1 DK1	Tuesday Tuesday Tuesday	9.00am - 10.00am 10.00am - 11.09am	WXES1101 WXES1401	Norisma, RJRY MKO
DK1 DK1 DK1 DK1 DK1	Tuesday Tuesday Tuesday Tuesday	9.00am - 10.00am 10.00am - 11.09am 11.00am - 12.00pm	WXES1101 WXES1401 WXES1401	Norisma, RJRY MKO MKO

Figure 4.14 View rooms' schedule window.

Chapter 5

# System Implementation and Testing



### 5.1 Developing Environment

Developing environment may have a certain impact on the development of a system. In order to speed up the system development, the suitable hardware and software should be used.

#### 5.1.1 Hardware

Hardware used for developing this automated scheduling system are listed as below:

- 350 MHz Pentium II Processor
- 64 MB RAM
- 3.2 GB Hard Disk Drive
- 14" color SVGA monitor
- CD-ROM Drive
- Floppy Drive
- Mouse
- Standard keyboard
- Canon Bubble Jet Printer 210-S

### 5.1.2 Software

Table 5.1 shows the software that have used in this project.

Software	Remarks
Visual Basic Version 6.0	This software was used in developing the system. All coding of the program involved is done with this software.
Microsoft Office	This software was used to prepare the proposal and report. It was also used to draw the entire diagram found in this report.
Capture Express 2000 Version 1.1	This software was used to capture the image of the screen of the application for the use of this report.

Table 5.1 Software used and descriptions.
## 5.2 System Development

Languages used to develop the automated scheduling system for FCSIT is Visual Basic 6.0. Visual Basic 6.0 is a user-friendly programming language. It is used to create all the user interfaces in the system.

In this system, Visual Basic is used to define the relationship between the objects involve in scheduling such as courses, students, lecturers, rooms, time and many others. Each of these objects may have a very complicated relationship between each other.

### 5.3 Database Development

Microsoft Access is used as the databases for the system. The database in the system is in mdb files form. All data can be added to the database directly as the application system is running or through Microsoft Access.

Module testing a converted from the design. Unit testing has been carried out ender a monitofield environment where a predictormined set of data has been provided to the modules. Unit testing has been carried out to observe what trant and output actions and the data produced (18)

in modulo testing, each of the sub-modules in the entries conties evoluties, reconcouries modules, locations entries modules, achodulable time modules, and execte

#### 5.4 Testing

Software testing refers to verification and validation of the program coded to solve the problems. Verification involves ensuring that the characteristics of a good design are incorporated into the program and the system is actually operates the way it was expected to be. Validation refers to execution of the program and system meets the requirements.

Therefore, the major focus in testing is to find the faults that might occurs in the application program. A test is successful only when a fault is discovered or when a failure has been come across. Testing actually involves the iteration of the process of fault identification and fault correction or removal.

In developing large system, testing usually involves several stages. These stages are module testing, integration testing, function testing, performance testing, acceptance testing and installation testing.

#### 5.4.1 Module Testing

Module testing or unit testing is to verify that the small unit functions properly with the types of input expected from the design. Unit testing has been carried out under a controlled environment where a predetermined set of data has been provided to the modules. Unit testing has been carried out to observe what input and output actions and the data produced [18].

In module testing, each of the sub-modules in the course entries modules, room entries modules, lecturers entries modules, schedulable time modules, and create schedule modules are tested separately. Ten each of the modules is tested for the creation of the user interface, the input data handling and output to data files, reset and exit from the module to make sure those modules does not actually has been design. Test cases have been developed to show that the input is properly converted to the desired output.

#### 5.4.2 Integration Testing

After the collections of the modules have been unit-tested, the next is to ensure that the interfaces among the components are defined and handled properly. Integration testing is the process of verifying that the system modules work together as describes in the system and program specification [18].

In this stage, all the individual sub-modules and modules are integrated and tested to ensure that the interfaces between the sub-modules and modules, modules and the main program are handled properly. Here all the small modules that are tested isolated before they are combined into a big program in the intelligent system and tested together.

The testing approach has been apply in the integration testing is the bottom-up integration. Each component at the lowest level of the system hierarchy is tested individually first. For the intelligent system, each sub-module is tested individually first, then the modules are tested. Finally, after the integration into a big program, the main program is test to ensure that system works correctly.

### 5.4.3 Function Testing

After the integration test, function test is carried out to assure that the system has the desired functionality. Function test will evaluate the system to determine if the functions described by the requirement specifications are actually performed by the integrated system [18].

The intelligent system in this stage is test to determine that it will schedule the courses and try to fix the courses to the schedule with the time and venue. Before, the system do so, it should be able to check all the constraints to make sure there is not any conflict to the schedule.

#### 5.4.4 Other Testing

The other testing should be carrying out after the functional testing are performance test, acceptance test and installation test.

Performance test compares the integrated components with the nonfunctional system requirements such as security, accuracy, speed and reliability. The users of the system to assure them that the system they need was the system that was built for them run acceptance test. Installation test allows users to exercise the system functions and document additional problems that result from being at the actual site [18].

These tests are not actually tested for the automated scheduling system for FCSIT because these tests should be carry out by the users of the system which may be the administration in the office.

Chapter



#### 6.1 System Evaluation

The system should be evaluated in order to know their effectiveness and efficiency of the system implementation. With system evaluation, the system can be improving by looking at the system limitations.

#### 6.1.1 System Strength

#### Simple interface

The user interfaces in this system are very simple. It is very easy to train the users on how to used the system.

#### Save time

The system has the capabilities of reducing the time and effort for generating the schedule.

#### Feasible schedule

The system produced schedule that was feasible and with sufficient quality to be used.

#### Speed

The system speeds up the generation of course schedule. The generation of schedule with the system is very fast and users do not need to spend much time on clerical work involve in scheduling. The schedule can be easily got from the data files that stored the schedule.

#### Dynamic and interactive content

Because the users provide all the data during the execution of the application, the content of the data are dynamic and interactive where the users can have their own codes and others for their data.

#### Automated scheduling process

The scheduling process is automated given the necessary input parameters such as the courses, lecturers, time and rooms.

#### 6.1.2 System Limitations

#### No error checking

There are no validations on user inputs into the data files. User may enter something that may affect the output of the system.

#### Limitation to the input

Input in the system should not include any characters that are not alphanumeric. Course structure can only take in one number.

#### Constraints integrated in the coding

The constraints on the course scheduling are integrated into the coding of the program and this limit the program ability to handle other constraints.

### **6.2 Future Enhancements**

Due to the limitations found in system, there is many improvements can be done as future enhancements on this system.

#### Web based

The application can be turn into a web based application in order to enable user to access the application from anywhere and the user only need a browser.

#### Updateable record

The record in the data files should be reusable and user can pick from the old data if they want to. This safe the users time from key in all the data all over again each time he uses the system.

#### Additional query functions

Application should provide other query functions than scheduling only. These functions include listing out the courses, lecturers, venues, certain type of courses and many more.

#### Apply the algorithms

The AI approach such as Genetic Algorithms, Memetic Algorithms, Simulated Annealing, Tabu Search, or Constraint Logic Programming can be integrated into the application to improve the scheduling process.

#### Flexible constraints

The users can provide any constraint to the application and the system should be able to process these constraints without integrated them into the codes for the system.

### • Object oriented

The concept of object oriented and reusable code should be given attention because this concept is very important to a better programming approach and the object-oriented concept is developing in the AI area.

### 6.3 Problem Encountered

There are many problems encounter during the system development. Most of these problems are due to the limitation in the language itself. Visual Basic is a very primitive language and therefore it cannot support certain type of functions or it needs to have longer code to execute the function.

Time given to develop the system is not long enough and yet many further improvements on the system cannot be done.

Debugging tool in Visual Basic is not powerful enough where the changes of the term and variables cannot be seen clearly like other language. This made the testing of the Visual Basic program time consuming.

### 6.4 Knowledge Gained

I'm provided with the opportunity to use the Visual Basic language to develop an AI application with this thesis. I had an opportunity to expose myself to this programming language.

In the process of completing this thesis, I learned more about the AI concepts such as rule-based, forward chaining, backward chaining, heuristic approach and also a little bit on Genetic Algorithms, Memetic Algorithms, Tabu Search and others.

### 6.5 Conclusions

The objective of the automated scheduling system is to provide an application that automated the process of scheduling in FCSIT is partially achieved with this application system.

The system actually can generate the schedule for the users but the schedule created is still not the best schedule and users are not given choice to choose from schedules. In addition to that, there are still many limitations found this intelligent system need to be improved in the future.

References

- [1] A. Apte, A. Jayasuriya, J. Kennington, I. Krass, R. Mohamed, S. Sorensen and J. Whitler, "Class Scheduling Algorithms for Navy Training Schools", Naval Research Logistics: An International Journal, Vol. 45 No. 6, September 1998, 533-551.
- [2] A. Bundy, "Artificial Intelligence Techniques: A Comprehensive Catalogue", 4<sup>th</sup> Revised Ed., Germany, Springer-Verlag Berlin Heidelberg, 1997.
- [3] A. Schaerf, "A Survey of Automated Timetabling", Centrum voor Wiskund en Infirmatica (CWI) report CS-R9567, The Netherlands, 1995.
- [4] A. Schaerf, "Tabu Search Techniques for Large High-School Timetabling Problems", Centrum voor Wiskund en Infirmatica (CWI) report CS-R9611, The Netherlands, 1996.
- [5] A. Wevell, "Word Power Dictionary", 1<sup>st</sup> Ed, The Reader's Digest Association South Africa (Pty) Ltd, 1996.
- [6] D. Corne, P. Ross and H.L. Fang, "Evolutionary Timetabling: Practice, Prospects and Work in Progress", UK Planning & Scheduling SIG Workshop, Strathclyde, September 1994.
- [7] E.K. Burke, D.G. Elliman and R.F. Weare, "A Hybrid Genetic Algorithm for Highly Constrained Timetabling Problems", The 6<sup>th</sup> International Conference on Genetic Algorithms, Pittsburgh USA, 15<sup>th</sup>-19<sup>th</sup> July 1995, 605-610.
- [8] E.K. Burke, D.G. Elliman and R.F. Weare, "The Automation of the Timetabling Process in Higher Education", Journal of Educational Technology Systems, Vol. 23(4), 1995, 257-266.

77

- [9] E.K. Burke, J.P. Newall and R.F. Weare, "A Memetic Algorithm for University Exam Timetabling", 1<sup>st</sup> International Conference on the Practice and Theory of Automated Timetabling, Napier University, Edinburgh, August 29<sup>th</sup>-September 1<sup>st</sup> 1995.
- [10] E.K. Burke, K.S. Jackson, J.H. Kingston and R.F. Weare, "Automated University Timetabling: The State of the Art", The Computer Journal, London, Vol. 40 No. 9, 1997, 565-571.
- [11] G. Lajos, "Complete University Modular Timetabling using Constraints Logic Programming", 1995.
- [12] G.F. Luger and W.A. Stubblefield, "Artificial Intelligence Structure and Strategies for Complex Problem Solving", 2<sup>nd</sup> Ed, California, The Benjamin/Cummings Publishing Company, 1993.
- [13] J. Durkin, "Expert System Design and Development", New York, Macmillan Publishing Company, 1994.
- [14] L.E. Frenzer, Jr., "Crash Course in Artificial Intelligence and Expert Systems", 1<sup>st</sup> Ed., USA, Howard W. Sams & Co..
- [15] P. Lucas and V.D.G. Linda, "Principles of Expert Systems", 1<sup>st</sup> printed, England, Addison-Wesley Publishing Company, 1991.
- [16] P. Ross, D. Corne and H.L. Fang, "Successful Lecture Timetabling with Evolutionary Algorithms", ECAI'94 Workshop W17: Applied Genetic and other Evolutionary Algorithms, 1994.
- [17] S. Deris, S.Omatu, H. Ohta and P. Saad, "Incorporating constraints propagation in genetic algorithm for university timetable planning", Engineering Application of Artificial Intelligence: The International Journal of Intelligent Real-time Automation, Vol. 12 No. 3, June 1999, 241-253.

[18] S.L. Pfleeger, "Software Engineering Theory and Practice", Washington, Prentice-Hall International, 1998.

# User Manual

### Installing Visual Basic 6.0

Step	Action
1	Insert the Visual Basic Version 6.0 CD into your CD-ROM drive.
2	Click on Start   Run on the Start Menu. The following window appears.
	Run
	Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.         Open:         OK       Eancel
3	Type x:\setup.exe (where x is your CD-Rom drive).
4	Following the step with clicking on Next to install Visual Basic.

## **Running Automated Scheduling System for FCSIT**



## Login Window

Step	Action
1	Click on System   Login or button on the toolbar. The following window appears.
	🛋 Log In
	User Name :
	Password :
	OK Cancel
2	Input the user name and password in the User Name: and Password: text fields.
3	ок
	Click on button to login to the system.
4	Click on Cancel button to close this screen.
	Click on an and the state of the state your password

Step	Action
1	Click on System   Change Password on the toolbar. The following windo appears.
	🖌 Change Password
	User Name :
	Old Password :
	New Password :
	Confirm Password :
	OK Cancel
2	Input the user name, old password, new password and confirm password in the User Name:, Old Password:, New Password: and Confirm Password: text fields.
3 (	Click onOK button to change your password.
4	Click on Cancel button to close this screen

## **Change Password Window**

## **Course Information Entries Window**

Ste	p Action
	Click on <b>Input   Course Information</b> or the button on the toolbar. The following window appears.
	S AUTOMATED SCHEDULING SYSTEM [Course Information Entries]
	System Input View Schedule About
	Course Code :
	Course Name
	Credit Hours : 3
	Students Size:
	Lecturers Attached: 1. GYes C No
	2, Course Type: Faculty
	Discipline : Computer Science
	Level: First Year
	Save Clear Cancel
	\$/24/01 4·23 PM
2	Input all the information and click on the Save button to save the data.
3	Clear button to clear all the information on the current window.
4	Click on the Cancel button to exit from this screen.

## **Lecturer Information Entries Window**

Ste	Action
	Click on <b>Input   Lecturer Information</b> or the <b>button</b> on the toolbar. The following window appears.
	& AUTOMATED SCHEDULING SYSTEM - [Lecture: Information Entries]
	Lecturer Code :
	Lecturer Name
	Courses Attached 1
	3
	Sava
	Dizaryot 4,21 FM
2	Input all the information and click on the Save button to save the data.
3	
	Click on the button to clear all the information on the current window.
4	Click on the Cancel button to exit from this screen.

## **Room Information Entries Window**

Ster	Action
	Click on <b>Input   Room Information</b> or the <b>button</b> on the toolbar. The following window appears.
	新 AUTOMATED SCHEDULING SYSTEM (Boom Information Entries)
	A System Input View Schedule About
	Room Name :
	Size Of Room :
	Room Type
e al le	🕫 Lacture Hall C Tutorial Room
	C Lecture Room C Computer Lab
	C Auditorium
	Save Clear Cancel
	9/24/01 4:24 PM
2	Input all the information and click on the Save button to save the data.
3	Click on the Clear button to clear all the information on the current window.
4	Click on the Cancel button to exit from this screen.

## **View Courses Information Window**

Step	Action
1	Click on View   Course Information or button on the toolbar. The following window appears.
	View By       Generctie List       Course Legend         Level:       All       Clear List       Concel         Cencel       If Frai Year       If Second Year         Selected Courses List       If Second Year       If Second Year         Course Code       Course Name       Credit Hours Size of students       Lecturers Attached
2	Select the discipline of the courses through the Discipline: drop down menu Discipline : All
3	Select the level of the courses through the Level: drop down menu Level: All



## **View Lecturers Information Window**

CI	ick on View   Lecture	r Information on the	
fol	llowing window appears	differentiation of thema button	on the toolbar.
-	AUTOMATED SCHEDULING SYSTEM - [L. System Input View Schedule About	ecturers Information]	
L.C.		18	-
	Lecturers List		
1	Lecturer Code Lecturer Name	Courses Attached	Generate Liet
			Clear List
			Cancel
			The second s
			8/24/01 4:20 PM
			8/24/01 4:20 PM
			8/24/01 4-25 PM
			8/24/01 4:20 PM
			8/24/01 4-25 PM
Chek			8/24/01 4:25 PM
Chai			8/24/01 4:25 PM
Chek		5	8/24/01 4:25 PM
Chek			8/24/01 4:25 PM
			8/24/01 4.25 PM
Chek			8/24/01 4:20 PM
Chek			8/24/01 4.25 PM

Step		Environment	Action	
2	Click on following	the Generate List g screen appears.	button to generate the le	ecturers' list.
	System Inp	D SCHEDULING SYSTEM [Lectures Into ut View Schedule Albout	rmation]	
	Lecturers	List		
1	Lecturer	Due Lecturer Name	Courses Attached	Generate List
	CTK	Abrizah Abdullah	WMET3102	Lastantantantantanta
	EH	En Chiew Thiam Kian	WXET1101	
	Fizik	In Fanzan Hanum Md Nasaruddin	WXET2103	Clear List
	HZ	Da Hana gaura Dal Caral	SMES2403	
	IB	En Ibrahim Abu Bahan	WRES3401,WXET1110	
	Kiran Keur	Kiran Keur	WRES3303, WXES2302	Cance!
	Law	Faculty of Law	WMET3106	and the second
	LSP	Prof Marchine Dal on Sai Dant	LXEB1303	
	Maizatul	Maizatul	WKES3202,WXES2201,WXES247	
	Matt	Jehetan Matamatik	WXE12103,WXET3204	
	Mat2	Johatan Matamatik	SJEWI102,SJEW2301	
	Media	Jabatan Media	SJEWITU2	
	МКО	En Mohd Khalit Othman	AKEA2124AKEA2316AKEA3308	
	ML	Pn Miss Leihe	WAES1401	
	NA	Cik Nor Aniza Abdullah	WDET2102	
	NEN	Nor Edzan Haji Nasir	WHE 13103	
	NF	NF	WVME12109	
	NMN	En Nonzaily Mohd Noor	WHE I 2102	
	NMY	Pn Norizan Mohd Yasin	WHEE 2201 WALE COORA	
	Norisma	Norisma	WAES2201 WAEC1101	
			8/24/01	4-25 PM
				1.4.4 J M
Cl	ick on the	Clear List bu	itton to clear the list.	*****
Cli	ck on the	Cancel	tton to exit from this scree	

## **View Rooms Information Window**

Step	Action
1	Click on View   Room Information or button on the toolbar. The following window appears.
	Rooms List       Room Code       Room Name       Size of Room       Room Type         H       Lecture Room         AUDI: Auditorium         TR:       Tutorial Room         LAB:       Computer Lab
	Generate List Clear List Cancel 8/24/01 4:23 PM

Step			Action		
2	Click on th following sc	Generate List creen appears.	button to	o generate t	he rooms' list. T
		HEDULING SYSTEM - IBooms Int	Courses and the second s		a the toolbox T
	Ca System Input Y	iew Schedule About	tomation		
				and the second	
	Boom Code	Deam Name	las so		Room Legend
	AUDI	Auditorium	Size of Room	Roam Type	
	BK1	Bilik Kuliah 1	100 60	AUDI	LH Lecture Hall
	BK2	Bilik Kullah 2	60	LR	LR: Lecture Room
	BK3	Bilik Kullah 3	60	LR	AUDI : Auditorium
	BK4 BK5	Billk Kultah 4	60	LR	TR: Tutorial Room
	DK1	Dewan Kullah 1	50	LR	LAB: Computer Lab
	DK2	Dewan Kuliah 2	300	LH	
	MM1	Makmal Mikro 1	50	LAB	
	MM2	Makmal Mikro 2	50	LAB	
	MM3	Makmal Boole	60	LAB	
	MM5	Makmal Ranokaian	60	LAB	
	MM6	Makmal Dilikstra	60	LAD	
		Senerate List	Clear List		Cancel
		ienerate List	Ciegr List		Cancel 8/24/31 4-28 PM
3	The acronym	enerate List	Clear List	s stated in the	Cancel B/24/01 4.28 PM e Room Legend.
3	The acronym	ienerate List	Clear List es of rooms as	s stated in the	Cancel B/24/01 4.28 FM e Room Legend.
3	The acronym	anerete List	Clear List	s stated in the	Cancel 8/24/01 4.28.PM e Room Legend.
3	The acronym Room Lege LH : Lec	senerate List	ClearList es of rooms as	s stated in the	Cancel 8/24/01 4.28 PM e Room Legend.
3	The acronym Room Lega LH : Lac LR : Lac	annerate List	ClearList es of rooms as	s stated in the	Cancel 8/24/91 4-28 FM C Room Legend.
3	The acronym Room Lege LH : Leo LR : Leo	annerate List	ClearList	s stated in the	Cancel 8/24/01 4:28 PM C Room Legend.
3	The acronym Room Lega LH : Lea LR : Lea AUDI : Aud	ionerate List	ClearList	s stated in the	Cancel 8/24/01 4.28 PM e Room Legend.
3	The acronym Room Lege LH : Lee LR : Lee AUDI : Aue TR : Tut	ienerate List	Clear List es of rooms as	s stated in the	Cancel B/24/01 4-28 FM e Room Legend.
3	The acronym Room Lega LH : Lea LR : Lea AUDI : Aud TR : Tut	ienerate List	ClearList es of rooms as	s stated in the	Cancel B/24/01 4.28 FM e Room Legend.
3	The acronym Room Lega LH : Lea LR : Lea AUDI : Aua TR : Tut LAB : Con	ienerate List	Clear List	s stated in the	Cancel B/24/01 4.28 FM e Room Legend.
3	The acronym FRoom Lega LH : Lea LR : Lea AUDI : Aua TR : Tut LAB : Con	ienerate List	Clear List	s stated in the	Cancel B/24/01 4.28 FM e Room Legend.
3	The acronym Floom Lege LH : Leo LR : Leo AUDI : Auo TR : Tut LAB : Con	annerente List	Clear List	s stated in the	Cancel B/24/01 4.28 FM e Room Legend.
3	The acronym Room Lege LH : Lee LR : Lee AUDI : Auo TR : Tut LAB : Con	annerente List	ClearList es of rooms as	s stated in the	Cancel 8/24/01 4:28 PM C Room Legend.
3	The acronym Room Lega LH : Lac LR : Lac AUDI : Auc TR : Tub LAB : Con	ionerate List	ClearList es of rooms as	s stated in the	Cancel B/24/01 4:28 FM e Room Legend.
3	The acronym Room Lega LH : Lea LR : Lea AUDI : Aua TR : Tut LAB : Con	anarate List	ClearList es of rooms as button to cl	s stated in the	Cancel B/24/01 4.25 FM e Room Legend.
3	The acronym Room Lega LH : Lea LR : Lea AUDI : Aud TR : Tut LAB : Con	senerate List	Clear List es of rooms as button to cl	s stated in the	Cancel B/24/01 4.28 FM e Room Legend.
3 4 5	The acronym Floom Lega LH : Lea LH : Lea LR : Lea AUDI : Aua TR : Tut LAB : Con	ienerate List	Clear List es of rooms as button to cl	s stated in the	Cancel B/24/01 4.28 FM
4	The acronym Floom Lege LH : Lee LH : Lee AUDI : Auc TR : Tut LAB : Con Click on the Click on the	ienerate List  as represent the type and thure Hall thure Room ditorium orial Room uputer Lab  Clear List  Cancel	ClearList es of rooms as button to cl	s stated in the lear the list.	Cancel B/24/01 4.28 FM e Room Legend.

## **Students Schedule Window**

1       Click on Schedule   Create Schedule or button on the toolt schedule will be generated.         2       Click on the Students Schedule tab and the following window appea         ••••••••••••••••••••••••••••••••••••	1       Click on Schedule   Create Schedule or button on the toolba schedule will be generated.         2       Click on the Students Schedule tab and the following window appears         2       Click on the Students Schedule tab and the following window appears         3       Students Schedule tab and the following window appears         4       Students Schedule         5       Students Schedule         1       Students Schedule         1       Course Legend         1       Students Schedule         1       S	1	Acuon
2       Click on the Students Schedule tab and the following window appea         2       Click on the Students Schedule tab and the following window appea         2       Students Schedule State About         2       Students Schedule         2       Students Schedule List         Course Code       Doy         3       Time         3       Select the discipline of the students through the Discipline: drop down         Discipline :       All         3       Select the level of the students through the Level: drop down         1       All	2       Click on the Students Schedule tab and the following window appears         AlloHATID Scheduling Schedule       Schedule         Students Schedule       Exclusers Schedule         Bludents Schedule       Exclusers Schedule         View By       Course Legend         Ucophine:       All         Course Course       Exclusers         Selected Students Schedule List       Course Course Attached         Course Code       Day         Time       Rooms Attached         Select the discipline of the students through the Discipline: drop down m         Discipline:       All	1	Click on Schedule   Create Schedule or button on the toolbar.
AllOMATED SCHEDULING SYSTEM - [Schedule]      Students Schedule / Students Schedule / Rooms Schedule / Rooms Schedule / Rooms Schedule / If Frat Yee      Selected Students Schedule List      Course Code Day Time Rooms Attached Lecturors Attack      Generate Schedule / Clear Cancel      972471 431P      Select the discipline of the students through the Discipline: drop down Discipline : All      Select the level of the students through the Level: drop down Level : All      Select the level of the students through the Level: drop down	All OMATIO Schedule System       Schedule         Pater pat yew schedule Apox       Rooms Schedule         Students Schedule       I resturers Schedule         View By       Discipline:         Discipline:       All         Selected Students Schedule       I secure and the students through the Discipline: drop down m         Orreents Schedule       Oregon         Generate Schedule       Oregon         Select the discipline of the students through the Discipline: drop down m         Discipline       All	2	Click on the Students Schedule tab and the following window appears
Openent park Meet Schedule       Students Schedule       Lacturors Schedule       Rooms Schedule         Students Schedule       I       Course Legend       If Fist Yes       If Fist Yes         Docphine:       All       If Second Yes       If Fist Yes       If Fist Yes         Selected Students Schedule List       Course Code       Day       Time       Rooms Attached       Lecturors Attached         Generate Schedule       Clear       Cancel       972401       431 P         Select the discipline of the students through the Discipline: drop down       Discipline :       All         Select the level of the students through the Level: drop down       Level:       All	Select the discipline of the students through the Discipline: drop down m         Discipline All		a Allowaten spicolitike system in
Students Schedule       Lecturers Schedule       Rooms Schedule         Discipline       All       Course Legend       If Fist Yes         Level:       All       Course Code       If Fist Yes         Selected Students Schedule List       Course Code       Day       Time         Generate Schedule       Clear       Cancel         9724/01       431 P         Select the discipline of the students through the Discipline: drop down         Discipline:       All	Students Schedule       Lecturers Schodule       Rooms Schedule         View By       Discipline:       All       Course Legend       If Fist Yes         Level:       All       Course Code       If Fist Yes       If Second Yes         Selected Students Schedule List       Course Code       Day       Time       Rooms Attached       Lecturers Attaches         Generate Schedule       Clear       Cancel       9724/01       4.31 Per         Select the discipline of the students through the Discipline: drop down m       Discipline :       All       Image: Select the level of the students through the Discipline in the model		B. System (nput View Schedule About
View By       Course Legend         Discipline:       II         Level:       II         Selected Students Schedule List       E Strid Year         Course Code       Day         Time       Rooms Attached         Level:       II The Year         Generate Schedule       Clear         Select the discipline of the students through the Discipline: drop down         Discipline:       All	View By       Course Legend         Discipline:       All         Level:       All         Selected Students Schedule List         Course Code       Day         Time       Rooms Attached         Level:       Day         Generate Schedule       Clear         Select the discipline of the students through the Discipline: drop down m         Discipline:       All		Students Schedule Rooms Schedule Rooms Schedule
Level:       All         Selected Students Schedule List       If Second Year         Course Code       Day         Time       Pooms Attached         Level:       All         Generate Schedule       Clear         Select the discipline of the students through the Discipline: drop down         Discipline:       All         Select the level of the students through the Level: drop down         Level:       All	Level: All Cieve Code Day Time Rooms Attached Lecturers Attaches Generate Schedule List Course Code Day Time Rooms Attached Lecturers Attaches Generate Schedule Cieve Cencel 9724/01 4:31 PM Select the discipline of the students through the Discipline: drop down m Discipline : All		View By Discipline : All Course Legend CS First Year IT First Year
Selected Students Schedule List         Course Code       Day         Time       Rooms Attached         Lecturers Attack         Generate Schedule       Clear         Variable       Clear         9724/01       431 P         Select the discipline of the students through the Discipline: drop down         Discipline :       All         Select the level of the students through the Level: drop down         Level :       All	Selected Students Schedule List         Course Code       Day         Time       Rooms Attached         Lecturers Attached         Generate Schedule       Clear         Select the discipline of the students through the Discipline: drop down m         Discipline :       All         Select the level of the students through the Level       Image: drop down m		Level: All CS Second Year If Second Year
Course Code       Dey       Time       Rooms Attached       Lecturers Attached         Generate Schedule       Clear       Cancel       9724/01       431 P         Select the discipline of the students through the Discipline: drop down       Discipline :       All       Image: All and	Course Code       Day       Time       Rooms Attached       Lecturers Attached         Generate Schedule       Clear       Cancel       9724/01       431 PH         Select the discipline of the students through the Discipline: drop down m       Discipline :       All       Image: Select the level of the students through the Level down m		Selected Students Schedule List
Generate Schedule       Clear       Cancel         9724/01       4.31 P         Select the discipline of the students through the Discipline: drop down         Discipline :       All         Select the level of the students through the Level: drop down         Level :       All	Generate Schedule Generate Schedule Clear 9724/01 4:31 PA Select the discipline of the students through the Discipline: drop down m Discipline: All		Course Code Day Time Rooms Attached Lecturers Attached
Generate Schedule       Clear       Cancel         8/24/01       4/31 P         Select the discipline of the students through the Discipline: drop down         Discipline :       All         Select the level of the students through the Level: drop down         Level :       All	Generate Schedule       Clear       Cancel         9724/01       4.31 PM         Select the discipline of the students through the Discipline: drop down m         Discipline :       All         Select the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the Level, down the level of the students through the students the students the students through the studen		
Select the discipline of the students through the Discipline: drop down Discipline : All	Select the discipline of the students through the <b>Discipline:</b> drop down m Discipline: All		
Select the discipline of the students through the Discipline: drop down Discipline : All . Select the level of the students through the Level: drop down Level : All	Select the discipline of the students through the <b>Discipline</b> : drop down m Discipline : All		Generata Schedule Clear Cancel
Select the level of the students through the Level: drop down	Select the level of the students through the Levels down to		Generate Schedule Clear Cancel 9724/01 4:31 PM
Level: All	mough the statements through the Level: drop down m	SE	Generate Schedule Clear Cancel 9724/01 4:31 PM elect the discipline of the students through the Discipline: drop down men iscipline : All
	Level: All	S	Generate Schedule       Clear       Cancel         9724/01       431 PM         elect the discipline of the students through the Discipline: drop down men         biscipline :       All         elect the level of the students through the Level: drop down men

Step			Acti	on			
5	Click on the	Generate : its. The foll	Schedule bu lowing screen	appears.	ate the schedule	of the	
	AUTOMATED SCHEDULING SYSTEM - ISchedulet						
	Spelen input View Schedule About						
	View By	Contraction of the second		Course Lopend			
	Discipline :	All	-	CS Fist Year	IT First Year		
				CS Second Year	IT Second Year		
	Level:	IAI	I	CS Third Year	IT Third Year		
	Selected Students	Schedule List				The second	
	Course Code	Day	Time	Rooms Atte	ched Lecturers Attac	ha	
	WXES1301 WXES1301	Tuesday	4.00pm - 5.00j	DK1	ZR.RM		
	WXES1301	Wednesday	8.00am - 9.00	am DK1	ZR.RM		
	WXES2103	Monday	3.00pm - 4.00p	om DK2	Nur Azlina, Rafi	da	
	WXES2103	Monday	5.00pm - 6.00p	om DK2	Nur Azlina, Rafi	da	
	WXES2201	Tuesday	8.00am - 9.00a	am DK2	LSP.Rafidah	aa	
	WAVE 9201	Tunnalmu	0.00 .00.00	and a second	An one of a source of the set is	101.01 20.074 44.4	
	WXES2201	Tuesday	9.00am - 10.00	lam DK2	LSP.Rafidah	1	
	WXES2201	Tuesday	9.00am - 10.00	Jam DK2	LSP.Rafidah	L-	
	WXES2201	Tuesday Schedule	9.00am - 10.00	Jam DK2	Cancel	L.	
	WXES2201	Tuesday Schedule	9.00am - 10.00	Jam DK2	LSP,Rafidah Cancel 8/24/01 4:33	E S	
	WXES2201	Tuesday Schedule	9.00am - 10.00 Clean	Jam DK2	Cancel 9/24/01 4:33	PM	
6	The colors repr	Tuesday Schedule	9.00am - 10.00 Clear different disc	Jam DK2	LSP.Rafidah Cancel 9/24/01 4:33 vels of the cour	PM	
6	The colors repr	Schedule resent the urse Lege	9.00am - 10.00 Clean different disc nd.	iplines and le	LSP.Rafidah Cancel 8/24/01 4:33 vels of the cour	PM Ses as	
6	The colors repr stated in the Co	Tuesday Schedule resent the urse Lege	9.00am - 10.00 Clean different disc nd.	Jam DK2	Cancel 9/24/01 4:33 vels of the cou	PM FM TSES as	
6	The colors repr stated in the Co	Tuesday Schedule resent the urse Legen	9.00am - 10.00 Clear different disc nd.	Jam DK2	LSP.Rafidah Cancel 8/24/01 4:33 vels of the cou	PM FSCS AS	
6	The colors repr stated in the Co	Tuesday Schedule	9.00am - 10.00 Clean different disc nd.	iplines and le	LSP.Rafidah Cancel 8/24/01 4:33 vels of the cour	PM	
6	The colors repr stated in the Co	Tuesday Schedule resent the urse Lege: nd ar	9.00am - 10.00 Clean different disc nd. IT First Year IT Second Yi	iplines and le	LSP.Rafidah	PM FSCS as	
6	The colors repr stated in the Co Course Leger CS First Ye CS Second	Tuesday Schedule resent the urse Lege: nd Par dYear	9.00am - 10.00 Clean different disc nd. IT First Year IT Second Year	iplines and le	LSP.Rafidah Cancel 9/24/01 4:33 vels of the cou	PM	
6	The colors repr stated in the Co Course Leger CS First Ye CS Second CS Third Ye	Tuesday Schedule resent the urse Lege nd ear dYear eat	9.00am - 10.00 Clean different disc nd. IT First Year IT Second Year IT Third Year	iplines and le	LSP.Rafidah Cancel 8/24/01 4:33 vels of the cour	PM FM	
6	The colors repr stated in the Co Course Leger CS First Ye CS Second CS Third Ye	Tuesday Schedule resent the urse Leges nd bar dYear ear	9.00am - 10.00 Clean different disc nd. IT First Year IT Second Year IT Second Year	iplines and le	LSP.Rafidah	PM PM	
6	The colors repr stated in the Co Course Leger CS First Ye CS Second CS Third Ye	Tuesday Schedule resent the urse Lege: nd ar d'Year ear	9.00am - 10.00 Clean different disc nd. IT First Year IT Second Year IT Third Year	iplines and le	LSP.Rafidah	PM FSCS AS	
6	The colors repr stated in the Co Course Leger CS First Ye CS Second CS Third Ye	Tuesday Schedule resent the urse Lege: nd ear d Year ear	9.00am - 10.00 Clean different disc nd. IT First Year IT Second Year IT Second Year IT Third Year ar	iplines and le	LSP.Rafidah	PM PM	
6	The colors repr stated in the Co Course Leger CS First Ye CS Second CS Third Ye CS Third Ye	Tuesday Schedule	9.00am - 10.00 Clean different disc nd. IT First Year IT Second Year IT Second Year IT Third Year ar	iplines and le	LSP.Rafidah	PM FSes as	
6	The colors repr stated in the Co Course Leger CS First Ye CS Second CS Third Ye CS Third Ye	Tuesday Schedule resent the urse Lege: nd ar dYear ear Cle	9.00am - 10.00 Clean different disc nd. IT First Year IT Second Year IT Second Year IT Third Year ar bu	iplines and le	LSP.Rafidah	PM FSes as	
6 7 8	The colors repr stated in the Co Course Leger CS First Ye CS Second CS Third Ye Click on the schedule list.	Tuesday Schedule resent the urse Lege: nd dYear eat Cle	9.00am - 10.00 Clean different disc nd. IT First Year IT Second Year IT Third Year ar bu	iplines and le	LSP.Rafidah	PM FSes as	
6 7 8	The colors repr stated in the Co Course Leger CS First Ye CS Second CS Third Ye CS Third Ye CS Third Ye	Tuesday Schedule resent the urse Leges nd dYear ear Cle	9.00am - 10.00 Clean different disc nd. IT First Year IT Second Year IT Second Year IT Third Year ar bu	iplines and le	LSP.Rafidah	PM FM FSES AS rses as	

### **Lecturers Schedule Window**

Nerp	Action					
1	Click on Schedule   Create Schedule or button on the toolbar. The schedule will be generated.					
2	Click on the Lecturers Schedule tab and the following window appears.					
	Students Schedule Lecturers Schedule Rooms Schedule					
	View By     Course Legend       Lecturer Code :     IT First Year       CS Second Year     IT Second Year       CS Third Year     IT Third Year					
	Selected Lecturer Schedule List           Lecturer Code         Day         Time         Course Attached         Boom Attached					
3	Generate Schedule Clear Cancel					
	8/24/01 4:34 PM					
3	Select the lecturer from the Lecturer Code: drop down menu Lecturer Code:					
3	Select the lecturer from the Lecturer Code: drop down menu Lecturer Code:					
3	9/24/01 4:34 PM Select the lecturer from the Lecturer Code: drop down menu Lecturer Code:					

Step		Action						
4	Click on the Generate Schedule button to generate the schedule of the selected lecturer. The following screen appears.							
2								
	Students Schedule	Lecturers Schedule	е [Р	Rooms Schedule				
	View By	Cour	se Legend					
	Lecturer Code : LSP		CS Second Year CS Third Year	IT First Year IT Second Year IT Thild Year				
	Selected Lecturer Schedule List							
	Lecturer Code Day	Time	Course Attached	Room Attached				
	LSP Monday	9.00am - 10.00am	WKES3202 WKES3202	DK1				
	LSP Monday	10.00am - 11.00am	WKES3202	DK1				
	LSP Tuesday	9.00am - 10.00am	WXES2201 WXES2201	DK2 DK2				
	LSP Monday LSP Monday	11.00am - 12.00pm	WXES2201	MM1				
	LSP Monday	11.00am - 12.00pm	WXES2201 WXES2201	MM2 MM3				
	Generate Schedule Olear Cencel							
				3/24/01 4:34 PM				
5	The colors represent the different district is							
	stated in the Course Legend.							
-	Course Legend							
	Lo First Year	IT First Year						
	CS Second Year	IT Second Year						
	CS Third Year	1T Third Year						
6	Click on the Clear schedule list.	button	to clear the	selected lecturer				
7	Click on the	button to	exit from this	screen.				

## **Rooms Schedule Window**

Step	Action						
1	Click on Schedule   Cr schedule will be generate	eate Schedule	or 🖄 butto	n on the toolbar. T			
2	Click on the Rooms Sche	edule tab and the	following w	indow oppose			
	AUTOMATED SCHEDULING SYSTEM - [Schedule]						
	Students Schedule	Lecturers Schr	idule Y	Booms Schedule			
	View By		Course Lesend	HDOME Schedule			
	Dunch		CS First Year	IT First Year			
	Moom Code ;		CS Second Year	IT Second Year			
				CS Third Year	T Third Year		
	Selected Room Schedule List		The second second				
	Room Code Day	Time	Course Attact	ed Lectorer Attached			
	Generate Schedule	Clear		Cancel			
	The second se	and and a second s		and the second of the second			
	I have been a second se						
			Ţ	0724/01 4:34 FM			
Step	Action						
------	---						
4	Click on the Generate Schedule button to generate the schedule of the selected room. The following screen appears.						
	AUTOMATED SCHEDULING SYSTEM [Schedule]      System (rpat View Schedule About      Content of the sector of th						
	Students Schedule Booms Schedule						
	View By Course Legend						
	Room Code :     DK1       CS Second Year     IT First Year       CS Second Year     IT Second Year       ES Third Year     IT Third Year						
	Selected Room Schedule List						
	Room Code         Day         Time         Course Attached         Lecturer Attached *           DK1         Mondey         3.00pm - 4.00pm         SJEW1182         Mat1.Mat2						
	DK1         Monday         4.00pm - 5.00pm         SJEW1102         Mat1.Mat2           DK1         Monday         5.00pm - 6.00pm         SJEW1102         Mat1.Mat2           DK1         Tuesday         8.00am - 9.00am         WXES1101         Norisma,RJRY						
	DK1         Tuesday         10.00am         WXES1101         Norisma.P.JRY           DK1         Tuesday         10.00am - 11.00am         WXES1401         MKO           DK1         Tuesday         11.00am - 12.00pm         WXES1401         MKO           DK1         Tuesday         12.00pm - 1.00pm         WXES1401         MKO						
	Generate Schedule Clear Cancel						
	8/24/01 4:42 PM						
5	The colors represent the different disciplines and levels of the courses as stated in the Course Legend.						
	Course Legend						
•	CS First Year IT First Year						
	CS Third Year IT Third Year						
6	Click on the Clear button to clear the selected room schedule list.						
7	Click on the Cancel button to exit from this screen.						

## **About ASS Window**

Step	Action
1	Click on About   ASS on the toolbar. The following window appears.
	About ASS
	Automated Scheduling System Version 1.0
	Faculty of Computer Science And Information Technology University of Malaya
2	Click on the button to exit from this screen.



## **Coding Sample**

Option Explicit Public dbLect As Database Public rsLect As Recordset Public dbRoom As Database Public rsRoom As Recordset Dim rows As Integer Dim j As Integer Dim z As Integer Dim countRow As Integer Dim pos As Integer Dim lect As String Dim lect1 As String Dim lect2 As String

Private Sub cmdClear1\_Click() cboDiscipline.Clear cboLevel.Clear flxStuSche.Clear flxStuSche.rows = 1 DisplayStuSche showcboDiscipline showcboLevel cboDiscipline.SetFocus End Sub

```
.....
```

Private Sub cmdCancel\_Click(Index As Integer) Unload frmSchedule End Sub

Private Sub DisplayStuSche() Dim i%

flxStuSche.Clear flxStuSche.rows = 1

flxStuSche.FormatString = "Course Code" + vbTab + "Day" + vbTab + "Time" + vbTab + \_\_\_\_\_ "Rooms Attached" + vbTab + "Lecturers Attached"

flxStuSche.Redraw = True flxStuSche.ColWidth(0) = flxStuSche.ColWidth(1) = flxStuSche.ColWidth(2) = flxStuSche.ColWidth(3) = flxStuSche.ColWidth(4) =

flxStuSche.Row = 0 For i = 0 To flxStuSche.Cols - 1 flxStuSche.Col = i flxStuSche.CellFontSize = 10 flxStuSche.CellFontBold = True flxStuSche.ColAlignment(i) = 1 flxStuSche.RowHeight(0) = 380 Next i End Sub Private Sub cmdGenerate1 Click() Dim displayMsg As String DisplayStuSche i = 0displayMsg = "Application is processing data " + vbCrLf + "This will take a few seconds. " + vbCrLf + "Please wait ..... " Call showProgress(30, 0, Int((10 \* Rnd) + 60), displayMsg) Me. MousePointer = 11If cboDiscipline. Text = "All" Then If cboLevel. Text = "All" Then For z = 0 To countRow - 1 **ShowCourse** If flx1.TextMatrix(z, 5) = "CS" And flx1.TextMatrix(z, 6) = "1" Then cColorLoop (vbRed) ElseIf flx1.TextMatrix(z, 5) = "CS" And flx1.TextMatrix(z, 6) = "2" Then cColorLoop (vbBlue) ElseIf flx1.TextMatrix(z, 5) = "CS" And flx1.TextMatrix(z, 6) = "3" Then cColorLoop (vbMagenta) ElseIf flx1.TextMatrix(z, 5) = "IT" And flx1.TextMatrix(z, 6) = "1" Then cColorLoop (vbBlack) ElseIf flx1.TextMatrix(z, 5) = "IT" And flx1.TextMatrix(z, 6) = "2" Then cColorLoop (vbGreen) ElseIf flx1.TextMatrix(z, 5) = "IT" And flx1.TextMatrix(z, 6) = "3" Then cColorLoop (vbCyan) End If Next z ElseIf cboLevel.Text = "First Year" Then For z = 0 To countRow - 1 If flx1.TextMatrix(z, 6) = "1" Then ShowCourse If flx1.TextMatrix(z, 5) = "CS" Then cColorLoop (vbRed) ElseIf flx1.TextMatrix(z, 5) = "IT" Then cColorLoop (vbBlack) End If End If Next z Elself cboLevel. Text = "Second Year" Then For z = 0 To countRow - 1 If flx1.TextMatrix(z, 6) = "2" Then **ShowCourse** If flx1.TextMatrix(z, 5) = "CS" Then cColorLoop (vbBlue) Elself flx1.TextMatrix(z, 5) = "IT" Then cColorLoop (vbGreen) End If End If Next z Elself cboLevel. Text = "Third Year" Then For z = 0 To countRow - 1 If flx1.TextMatrix(z, 6) = "3" Then ShowCourse If flx1.TextMatrix(z, 5) = "CS" Then cColorLoop (vbMagenta)

```
ElseIf flx1. TextMatrix(z, 5) = "IT" Then
               cColorLoop (vbCyan)
             End If
          End If
       Next z
     End If
  Elself cboDiscipline. Text = "Computer Science" Then
     If cboLevel. Text = "All" Then
       For z = 0 To countRow - 1
          If flx1.TextMatrix(z, 5) = "CS" Then
            ShowCourse
            If flx1.TextMatrix(z, 6) = "1" Then
               cColorLoop (vbRed)
            ElseIf flx1. TextMatrix(z, 6) = "2" Then
              cColorLoop (vbBlue)
            ElseIf flx1. TextMatrix(z, 6) = "3" Then
              cColorLoop (vbMagenta)
           End If
         End If
      Next z
    Elself cboLevel.Text = "First Year" Then
      For z = 0 To countRow - 1
         If flx1.TextMatrix(z, 5) = "CS" And flx1.TextMatrix(z, 6) = "1" Then
           ShowCourse
           cColorLoop (vbRed)
        End If
      Next z
   Elself cboLevel. Text = "Second Year" Then
      For z = 0 To countRow - 1
        If flx1.TextMatrix(z, 5) = "CS" And flx1.TextMatrix(z, 6) = "2" Then
           ShowCourse
          cColorLoop (vbBlue)
        End If
     Next z
   Elself cboLevel. Text = "Third Year" Then
     For z = 0 To countRow - 1
       If flx1.TextMatrix(z, 5) = "CS" And flx1.TextMatrix(z, 6) = "3" Then
          ShowCourse
          cColorLoop (vbMagenta)
       End If
     Next z
  End If
ElseIf cboDiscipline. Text = "Information Technology" Then
  If cboLevel. Text = "All" Then
     For z = 0 To countRow - 1
       If flx1. TextMatrix(z, 5) = "IT" Then
         ShowCourse
         If flx1.TextMatrix(z, 6) = "1" Then
           cColorLoop (vbBlack)
        Elself flx1. TextMatrix(z, 6) = "2" Then
          cColorLoop (vbGreen)
         ElseIf flx1.TextMatrix(z, 6) = "3" Then
           cColorLoop (vbCyan)
         End If
      End If
```

```
Next z
        Elself cboLevel. Text = "First Year" Then
          For z = 0 To countRow - 1
         If flx1.TextMatrix(z, 5) = "IT" And flx1.TextMatrix(z, 6) = "1" Then
               ShowCourse
               cColorLoop (vbBlack)
            End If
          Next z
       Elself cboLevel. Text = "Second Year" Then
         For z = 0 To countRow - 1
            If flx1.TextMatrix(z, 5) = "IT" And flx1.TextMatrix(z, 6) = "2" Then
              ShowCourse
              cColorLoop (vbGreen)
            End If
    Next z
      Elself cboLevel. Text = "Third Year" Then
        For z = 0 To countRow - 1
           If flx1.TextMatrix(z, 5) = "IT" And flx1.TextMatrix(z, 6) = "3" Then
             ShowCourse
             cColorLoop (vbCyan)
           End If
        Next z
      End If
   End If
   Me. MousePointer = 0
   displayMsg = "Process will be completed soon." + vbCrLf + "Please wait ..... "
   Call showProgress(20, 68, 100, displayMsg)
End Sub
.....
Private Sub Form_Load()
Dim displayMsg As String
Dim message As String
Dim title As String
  displayMsg = "Application is processing data " + vbCrLf + "This will take a few seconds. " +
vbCrLf + "Please wait ..... "
  Call showProgress(30, 0, Int((10 * Rnd) + 20), displayMsg)
  Me.MousePointer = 11
 Call Schedule
  Me. MousePointer = 0
 displayMsg = "Process will be completed soon." + vbCrLf + "Please wait ..... "
 Call showProgress(20, 68, 100, displayMsg)
 showcboDiscipline
 showcboLevel
 showcboLectCode
 showcboRoomCode
 DisplayStuSche
 DisplayLectSche
 DisplayRoomSche
```

End Sub

Private Sub showcboLectCode() Dim i As Integer Dim LectCode As String

Set dbLect == OpenDatabase(App.Path + "\databases\lect.mdb") Set rsLect == dbLect.OpenRecordset("Lecturers", dbOpenTable)

```
rows = 0
rows = rsLect.RecordCount
cboLectCode.Clear
For i = 1 To rows
LectCode = rsLect.Fields("Lecturer Code").Value
cboLectCode.AddItem LectCode
rsLect.MoveNext
Next i
```

## End Sub

Private Sub ShowCourse() Dim cDay As String Dim cTime As String Dim cCou As String Dim cRoom As String Dim cLect As String Dim s As String

> cDay = flx1.TextMatrix(z, 0) cTime = flx1.TextMatrix(z, 1) cCou = flx1.TextMatrix(z, 2) cRoom = flx1.TextMatrix(z, 3) cLect = flx1.TextMatrix(z, 4)

s = cCou + vbTab + cDay + vbTab + cTime + vbTab + cRoom + vbTab + cLectflxStuSche. AddItem s

j = j + 1

-----

End Sub

```
Private Sub cColorLoop(colorCode)
Dim i As Integer
Dim k As Integer
```

```
flxStuSche.Row = 1

For i = 1 To flxStuSche.rows - 1

If i = j Then

For k = 0 To flxStuSche.Cols - 1

flxStuSche.Col = k

flxStuSche.CellForeColor = colorCode

flxStuSche.CellFortBold = True

Next k

Else

flxStuSche.Row = flxStuSche.Row + 1

End If

Next i

End Sub
```