FESP

Final Exam Scheduling Package

By
Kamal A. Haidar

May 1995

FESP

Final Exam Scheduling Package

By
Kamal A. Haidar

THESIS/PROJECT WORK

Submitted in partial fulfillment of the requirements for the degree of
Master of Science in Computer Science
at the Lebanese American University
May 1995

[redacted]

Dr. Nashat Mansour (Advisor)
Assistant Professor of Computer Science
Lebanese American University

[redacted]

Dr. Walid Keirouz
Assistant Professor of Computer Science
Lebanese American University

[redacted]

Dr. George Nasr
Assistant Professor of Electrical and Computer Engineering
Lebanese American University

ABSTRACT

Final Exam Scheduling is a real and troublesome problem that all institutions of higher education are faced with a few times a year. This report presents the implementation of a **F**inal **E**xam **S**cheduling **P**ackage (**FESP**) at the LEBANESE AMERICAN UNIVERSITY. **FESP** distributes the final exams among the exam sessions and assigns exams to classrooms. It takes into consideration student conflicts and accommodates splitting of exams among classrooms and sharing classrooms among exams. We present the package specification, the approach, the algorithms, the design, and the implementation techniques (using **ORACLE R-DBMS** and **Tools**). The experimental results obtained on real-life data show that **FESP** yields a substantial decrease in the number of exam conflicts in comparison with those obtained by the manually prepared exam schedule.

ACKNOWLEDGMENTS

I would like to express my gratitude to Professor Nashat Mansour, my advisor, for the encouragement, help and support he gave me. I extend my gratitude to the second readers, Professor George Nasr and Professor Walid Keirouz.

I am very grateful to the Harriri foundation and LOGOS Company for their financial support. Special thanks go to the Academic and the Administrative Computer Centers' Staff for their cooperation. I greatly thank Professor Wadih Juriedini for the valuable discussions and I express my deep respect and appreciation for him.

I would like to thank all those who assisted me to accomplish this work, in one way or another.

Last, but not least, I feel honored to dedicate this work to the person who gave me the great support to achieve what I have achieved, Miss Amal Abi Faris.

TABLE OF CONTENTS

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

Final exams scheduling involves the assignment of the final exams to days, periods and classrooms in such a way that satisfies several conditions (constraints). In this project, the aim is building a scheduling software package that automates the scheduling process to produce a high-quality schedule in a reasonable execution time.

Exam scheduling is difficult to be done manually and efficiently for several reasons. Real instances can be very large with thousands of students registered in hundreds of courses (*i.e.* hundreds of final exams). The schedule is supposed to squeeze these final exams (at the end of the semester) into an exam period consisting of a limited number of days. These exams should also be distributed among a limited number of classrooms with different capacities.

For instance, during Spring 94-95, LEBANESE AMERICAN UNIVERSITY (LAU) had about 10,000 student-exam, a 6-day examination period, about 850 classroom seating capacity.

The scheduling of final exams has been defined as the process of minimizing three types of conflicts, in addition to the optimal classroom assignment. The first conflict indicates the number of students having simultaneous exams, the second indicates the number of

students with more than two exams per day, and the third indicates the number of students with consecutive exams.

The FINAL EXAMS SCHEDULING PACKAGE **(FESP)** described in this report is based on a mathematical model presented by Lotfi and Cerveny [Lotfi and Cerveny 1991]. The model consists of several mathematical formulations for minimizing the three conflicts mentioned above. Each formulation is associated with a heuristic algorithm to solve it. Certain major variations from the Lotfi and Cerveny solution have taken place in **FESP**. First, since the aim was to fit the LEBANESE AMERICAN UNIVERSITY rules and regulations concerning final exams scheduling, some new conditions were added to the heuristics. Second, Lotfi and Cerveny did not present a formulation and a heuristic for solving the problem of distributing the exam groups within or among exam days, so we introduce both the formulation and the heuristic to solve this problem. Third, we make it possible for the users to enforce some exam assignments (USER INTERVENTION) and assign the reading days before the scheduling process. Finally, we allow the splitting of one room among several exams during the same session (we will refer to an exam session by the term period, all through this report).

The implementation took place at the LAU Academic Computer Center using **ORACLE** (V6) R-DBMS and tools under DOS (PC Platform).

This report is organized as follows. Chapter 2 defines the problem and the detailed specification of the package. In Chapter 3, we explore the solution approach and the algorithms used in tackling the given problem. Chapter 4 addresses the design and implementation issues of **FESP**. In Chapter 5, we report experimental results in which we have used various test cases including the ones involving actual data from LAU. In Chapter 6 we present some related work in this field. In Chapter 7, we present our conclusion and propose some further work .

CHAPTER 2

SPECIFICATIONS AND GOALS

This chapter presents the problem specifications and goals. The problem of scheduling can be defined as follows: (a) Final exams assignment to exam periods within exam days, (b) Classroom assignment.

What are the objectives behind the development and the implementation of **FESP**? **FESP** aims at producing a final-exam schedule that minimizes the three types of conflicts associated with this problem: (1) students with simultaneous exams (direct conflict), (2) students with more than two exams per day, (3) students with back-to-back exams (consecutive exams).

1. It is acceptable to have a student with a direct-conflict on one condition: The conflict should result between a course and its prerequisite if the student is taking both (Newly introduced by **FESP**). However, there is a parameter to **FESP** that allows the activation or de-activation of such a condition. By default, it is deactivated.

2. It is acceptable to assign two exams given by the same instructor to the same period within the same day.

3. It is assumed that the number of days, periods within days, and classrooms' capacity fit each other *i.e.* students assigned to one period should have enough seats.

4. Last session of one day is considered as back-to-back with the first session of the next day.

5. There is no limit on the number of periods within each day. This number should be specified. This number is unique for all examination days.

6. There is no limit on the number of exam days. This number should be specified. Reading Days should be specified.

7. Classrooms that are to be used during examinations should be specified. A room may hold several sections (newly introduced by **FESP**), and a section might be split into more than one room.

8. The user can assign an exam to a day/period/room before scheduling. Such an enforced assignment will not be affected by **FESP**. Such forced scheduling is the only user-intervention means to **FESP** (newly introduced by **FESP**).

9. The total seating capacity of all the classrooms must be RC times the total number of students in a given period. RC is the ratio of classroom capacity to the number of students to be scheduled into it.

10. The interface means between **FESP** and the software package used in the institution is considered to be an auxiliary program that have been run and all data needed for **FESP** are already prepared. Such a program will be dependent on the software used by the given institution. **FESP** provides a sample program that interfaces with ORACLE-based applications.

Regarding, the goals **FESP** is trying to achieve, it was agreed that any scheduling package had to corporate the following set of characteristics [Lotfi and Cerveny 1991]:

- **Solution Quality:** Quality is to be measured in terms of the number of students having various types of conflicts prioritized according to the following creteria:
    1. Number of students with simultaneous exams
    2. Number of students with two or more exams per day
    3. Number of students with consecutive exams

The package has to produce a good-quality schedule. i.e. the goal is to minimize the conflicts presented above.

- **Efficiency:** Efficiency is measured in terms of core and disk storage space requirements and speed. The package should need the least amount of these resources.

- **Flexibility:** The package should be flexible enough to provide for changes in the input variables (*e.g.*, number of exam days, number of periods per day).

- **Ease of Maintenance:** The package is supposed to be easy for maintenance and running.

- **User Friendliness:** Input data is supposed to be easy to prepare and the output easy to understand.

CHAPTER 3

SCHEDULING APPROACH AND HEURISTICS

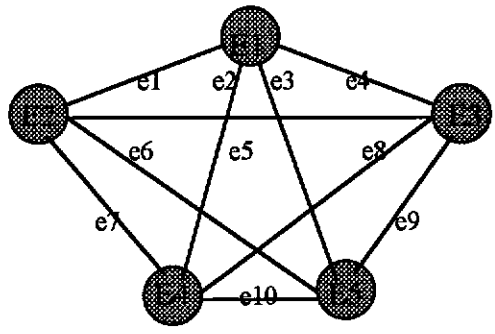
The process of scheduling was always viewed as a set of sub-processes (phases) each solving a sub-problem (phase objective) reaching to a general solution for the whole process (a schedule).

The proposed scheduling process consists of four phases [Lotfi and Curveny 1991]. In the first phase, all of the final exams are grouped into sets called blocks to minimize the direct conflicts. The second phase consists of assigning exam blocks to exams days to minimize the number of students with more than two exams per day. The third phase involves the arrangement of exam days as well as exam blocks within exam days to minimize the number of students with consecutive exams (last session of one day is considered consecutive to the first session of the next day). In the fourth phase, the exams are assigned to classrooms so as to maximize the space utilization. During any phase, any pre-fixed (User Intervention) sub-schedule should not be affected (Newly introduced by **FESP**).

In this chapter we present the problem formulation and the heuristic procedures for each of the four phases. Chapter 4 will highlight newly introduced features and variations in **FESP** from the proposed solution.

1. Assigning Exams to Exam Blocks (*Phase I*)

Grouping exams in blocks will make the day/period assignments of exams much easier. Instead of trying to assign hundreds of exams to periods within the exam days, we will be assigning tens of exam groups to these periods. This is acceptable if the grouping mechanism insures that each block contains only independent exams with no intersections among them.



The nodes (E_i) represent the final exams to be scheduled in a given semester. The edges represent any intersection between these nodes (exams). The weight (e_i) associated with each edge resembles the degree of the intersection *i.e.* the number of students sharing the two exams connected by the given edge.

Figure 3.1. Graph representation of the final exams.

1.1 Formulation

This problem is described by utilizing graph theory. Let G(V,E) be a graph (Fig. 3.1) in which each vertex $v \in V$ represents a final exam, and let $N = |V|$ be the number of final exams. An edge $(i,j) \in E$ represents the existence of at least one student taking both exams i & j. Associate with each edge $(i,j) \in E$, a_{ij} = number of students participating in exams i & j. Let $a_{ij} = 0$ for $(i,j) \notin E$. Two vertices i & j are said to be adjacent if the edge $(i,j) \in E$. (*i.e.*, $a_{ij} > 0$). The problem described in this phase is formulated as a Quadratic Assignment Problem. Table 3.1 shows the mathematical formulation of this problem.

Let $x_{ij} = 1$ if exam i is assigned to block j and 0 otherwise.

$$\min Z = 2Z' = \sum_{j=1}^{P} \sum_{i=1}^{N} \sum_{k=1}^{N} a_{ik} x_{ij} x_{kj}$$

subject to

1. $\sum_{j=1}^{P} x_{ij} = 1$ for $i = 1,2,\ldots,N$
2. $\sum_{i=1}^{N} x_{ij} \leq Q$ for $j = 1,2,\ldots,P$
3. $x_{ij} + x_{kj} \leq 2 - b_{ik}$ for $i,k = 1,2,\ldots,N$ & $j = 1,2,\ldots P$.

Where
- N = number of final exams
- P = number of available blocks
- a_{ik}= number of students taking exam i & k.
- $x_{ij} = 0,1$ for $\forall$ i,j (integer decision variables)
- $b_{ik} = 1$ if exam i and exam k have the same instructor & 0 otherwise.

Table 3.1. QAP Mathematical formulation of phase I.

In the above formulation, Z' is interpreted as the total number of students having simultaneous exams and is to be minimized. Such minimization is constrained in accordance with the given institutional rules and with some general controlling conditions. To ensure that each exam is assigned to one and only one block, constraint (1) is added. Constraint (2) forces the number of exams assigned to the same block to be less than or equal to Q, a function of the seating capacity available in the institution during examination period. Constraint (3) prevents two exams with the same instructor from being scheduled simultaneously. Note that a limited number of blocks P is available. The value of P depends on the number of exam days, the Q value, and the number of students registering during the given semester. However, some institutions have regulations that might violate these constraints, for instance LAU regulations violate the third constraint, besides, it allows students with simultaneous exams if one of these exams is prerequisite of the other, leaving for the unit/division concerned to decide on how to resolve such conflicts.

1.2 Phase I Heuristic Procedure

All of the final exams (nodes of G) are sorted in descending order of weighted degrees and saved in a list. The weighted degree of a node is defined as: $w_i = d_i \sum a_{ik}$, where d_i is the degree of node i and $\sum a_{ik}$ is the total number of students taking exam i and all other exams in conflict with exam i.

The reason for using the weighted degree is that it incorporates both the number of students and the number of exams in conflict with it to give a sense of difficulty of scheduling it. Therefore, a coloring algorithm schedules the more difficult (more conflicting) exams first.

Step-0:

1. Sets the status of all blocks to be assigned to final exams as empty blocks; i.e. available blocks.

2. The top item-exam in the ordered list is selected and assigned to the first available block (B). (Note: all blocks have just been initialized). Exam list is adjusted to account for the assigned exam, in order not to assign the same exam to more than one block.

3. A set-variable (V) is utilized to keep track of overloaded blocks. (Later the job of (V) will be clarified). (V) is initialized to (B).

4. The block counter of (B) is incremented.(Note: later on we will check that no block will be assigned more than a specified upper limit).

STEP-1:

1. Check if the exam list is not empty i.e. check whether the process is done or not. (Note: in the trivial case, where one exam is being scheduled, the process is done at **STEP-0**).

2. If the exam list is exhausted, then move to **STEP-4**. Otherwise, the top-item exam (A) of the list is fetched and the following is done:

Potential unfilled blocks (i.e. blocks that are able to include the given exam satisfying the conflicts' conditions) are grouped. If the group happened to be empty, i.e. the given course (A) conflicts with at least one course in each unfilled block, then move to **STEP-2**. Otherwise, i.e. there exists at least one potential unfilled block (B), assignment of (A) is done to (B), the block counter of (B) is incremented, exam list is updated, and the same step is repeated (**STEP-1**).

STEP-2:

Exam (A) was not assigned to any block because of either student or instructor type conflicts or both. Check to see if any non-potential unfilled block is still available. If so, move to **STEP-3**, otherwise, do the following:

A new set of potential unfilled blocks (Now,. blocks that are able to include the given exam satisfying only the instructor-type conflicts' conditions) are grouped. If the group happened to be empty, i.e. the given course (A) conflicts with at least one course in each unfilled block (instructor-type conflict), then no alternative is suggested by the heuristic. Otherwise, i.e. there exists at least one potential unfilled block, assignment of (A) is done to the first potential unfilled block (B) in this group, the block counter of (B) is incremented, exam list is updated, and move back to **STEP-1**.

STEP-3:

Exam (A) was not assigned to any block because of conflicts. The unfilled non-potential block (b) that is assigned the minimum number of exams is selected. (A) is assigned to this block, this block is flagged as overloaded, (V) is incremented, exam list is updated, and move back to **STEP-1**.

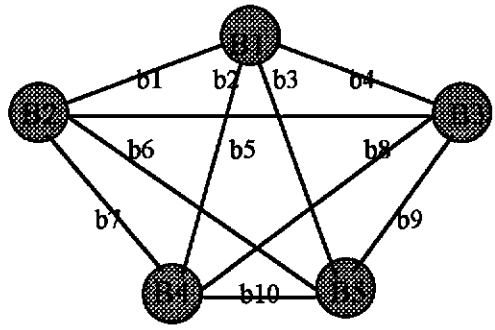
STEP-4:

When all courses are assigned to blocks. The list of blocks associated with the assigned exams will be ready as the input to phase II. The heuristic suggests to apply the revised version of Carlson's method on this solution for possible improvements.

2. Assigning Exam Blocks to Exam Days (*Phase II*)

After grouping all exams into several blocks, these blocks need to be assigned to periods within the exam days. The assignment mechanism's main concern is to have all blocks assigned in such a way that minimizes students with more than two exams per day. It is worth to note at this point that the trivial optimal mechanism is to assign one and only one block to each exam day. Practically, this is not acceptable since a limited number of exam days are to be used only.

One way to assign the blocks to exam days is to assign the blocks sequentially, starting with the first exam block. Such an approach, which in fact has been used by several existing scheduling packages, usually produces undesirable schedules in terms of the second- and third level conflicts. Another approach formulates this phase as a TRAVELING SALESMAN PROBLEM (TSP). Such formulation assumes that the last period of one day is adjacent to the first period of the next day. In this formulation, the cities represent the exam blocks and an optimal salesman tour constitutes the arrangement of the P exam blocks with a minimum number of students having consecutive exams.



The nodes (B_i) represent the exam blocks to be distributed among the exam days of the scheduling period in a given semester. The edges represent any intersection between these nodes (blocks). The weight (b_i) associated with each edge resembles the degree of the intersection i.e. the number of students having exams in the two blocks connected by the given edge. The problem becomes similar to that of phase I in the following sense: (a) Instead of grouping exams into blocks, blocks will be grouped into sets (days). (b) Days (new sets) should contain blocks with 'minimum' intersection (students with two exams per day cases are allowed to occur).

Figure 3.2. Graph representation of the Exam Blocks.

2.1 Formulation

In this work, the problem of assigning exam blocks to exam days is being formulated as a Quadratic Assignment Problem (QAP) with column constraints. Table 3.2 presents this formulation.

In this formulation, s' represents the number of students with two or more exams per day and to be minimized. This optimization problem is subject to two main constraints: (1) Every block should be assigned to one and only one day, (2) A limited number of blocks is to be assigned to the same day.

Let $y_{ij} = 1$ if block i is assigned to day j and 0 otherwise.

$$\min s = 2s' = \sum_{j=1}^{D} \sum_{i=1}^{P} \sum_{k=1}^{P} c_{ik}\, y_{ij}\, y_{kj}$$

subject to

1. $\sum_{j=1}^{D} y_{ij} = 1$ for i = 1,2,...,P
2. $\sum_{i=1}^{P} y_{ij} = PD_j$ for j = 1,2,...,D

Where D = number of exam days
PD_j = number of available blocks in day j
c_{ik} = number of students with exams in both blocks i & k (i <> k)
$c_{ii} = 0$ for all i.

Table 3.2. QAP Mathematical formulation of phase II.

2.2 Phase II Heuristic Procedure

The algorithm begins by constructing an initial solution obtained by assigning the exam blocks to exam days sequentially. That is, the first P/D blocks are assigned to the first day, the second P/D to the second day and so on. Then the current solution is examined for possible improvements through a pair-wise exchange of exam blocks. If no further improvement is possible then the algorithm stops and the current solution is used as input to the third phase.

To begin, use a sequential assignment to construct an initial solution. Let q = P/Q, and

$y_{ik} = 1$ for $i = (k-1)*q+2, \ldots, (k-1)*q+q(k-1)$ and $k = 1,2,\ldots D$.
$= 0$ otherwise.

Step-0:

1. C[c_{ij}] and Y[y_{ij}] matrices are used to determine R = CY; remind you that c_{ij} is the number of students with exams in both blocks i & j, y_{ij} is 1 if block i is assigned to day j, 0 otherwise.

2. Matrix R is examined row by row to get the minimum value in each.

3. The row and column indices where the minimum was found are checked for possible assignment of block (row index) to day (column index) after certain computations in **STEP-2.**

4. Initialize a certain FLAG to False (If FLAG remains False then no further improvements are possible).

5. The row index (i) that will be used to scan matrix R is initialized to 0.

STEP-1:

1. Move to **STEP-4** if all R-matrix rows are scanned.

2. Otherwise, get the minimum value (T) and column value (J1) in R-matrix i-row.

3. If block (i) is not assigned to day (j) then move to **STEP-2** (Possible assignment of block (i) to day (j), i.e. we detect a block (i) that has a minimum number of conflicts with blocks assigned to day (j)).

4. If this block is assigned to day (j) then the same step is repeated (**STEP-1**) (i.e. try another R-matrix row for possible improvements).

STEP-2: (i.e. a block may be assigned to another day)

1. Find in Y-matrix row (i) (i is the R-matrix row index in hand) for the day (t) assigned to block (I).

2. For every row j in Y-matrix column q = (J1) compute the following:

$$w_j = (r_{it} - r_{iq}) + (r_{jq} - r_{jt}).$$

3. Get the min w_k. If $w_k <= 0$ then move to **STEP-1,** otherwise to **STEP-3.**

STEP-3:

1. Interchange the assignment of (i) and (k) to days (q) and (t) (i.e. we assign blocks (i) and (k) to days (q) and (t) respectively, and cancel the assignment of blocks (k) and (i) to days (q) and (t) respectively).

2. Re-compute R = CY, set the Flag to 'True' and move to **STEP-1**.

In this step an improvement was made on the starting solution by interchanging the assignment of blocks to days.

STEP-4, (i.e. if all R-matrix rows were scanned (i) > P)

1. Check the Flag value. If False then this phase comes to an end (Flag = False ==> No improvements were done during the last iteration since **STEP-3** was not reached), otherwise, (i.e. Flag = True and some improvements were done) set Flag to 'False', re-initialize R-matrix index (i) to 0 and go back to **STEP-1**.

3. Arranging Exam Days and Blocks within Days (*Phase III*)

Phase II assigned exam blocks to exam days in a way to minimize the number of students with more than two exams per day. Phase III arranges these blocks within days to minimize the number of students with consecutive exams in the same day and arranges the exam days to minimize the number of students with consecutive exams between days. In

this work, we consider that the last period of one day is back-to-back to the first period of the next day. Phase III formulation and heuristic procedure are newly introduced by **FESP**.

It is worth noting at this point that the trivial case of arranging exam blocks within days is when R = D/P (the number of blocks to be assigned per day) equals to 3. Some institutions use this trivial value. When R is less than 3, there is nothing to optimize in the first part of phase III. On the other hand, the trivial solution to arrange exam days so as to minimize the second type of back-to-back conflicts is to insert a reading day after each exam day. This trivial case is not accepted in general by any institution.

3.1 Formulation and Heuristic Procedure

The phase III problem is sub-divided into two parts in accordance with the types of conflicts it attempts to tackle: In this work, we consider the following two types of consecutive exams: (1) Back-to-back exams in the same day, (2) Last period of one day is back-to-back to the first period of the next day. In both parts we formulate the solution as a TSP referring to some greedy [Brassard and Bratley 1988] and exact methods.

3.1.1 Within exam days

In this back-to-back type, the cities represent the exam blocks. An optimal salesman tour within each exam day constitutes the arrangement of the R blocks with a minimum number of students having consecutive exams. In other words, consider each exam day as being a complete undirected graph (G) where the nodes represent the exam blocks and the weight on each edge (i,j) of (G) represents the intersection (conflict) between blocks i and j. In fact, R is usually less than 5, consequently (G) will be a small graph that can be exhaustively searched for the minimum salesman tour. Thus, an optimal solution for this type of conflicts can be achieved in a reasonably accepted time. The formulation of the given problem is presented in Table 3(a).

3.1.2 Between exam days

In this back-to-back type, the cities represent the exam days. An optimal salesman tour within each exam day constitutes the arrangement of the D exam days with a minimum number of students having consecutive exams between blocks B_R of day D_j and B_1 of day D_{j+1}. In other words, consider the whole problem as being a complete directed graph (K) where the nodes represent the days and the weight on each edge (i,j) of (K) represents the intersection (conflict) between blocks B_R of day D_j and B_1 of day D_{j+1}. In fact, D usually ranges between 4 and 12, consequently (K) will be considered as a small graph that can be exhaustively searched for the minimum salesman tour as in the case of (G) above. Thus, an

optimal solution for this type of conflicts can be achieved in a reasonably accepted time.

The formulation of the given problem is presented in Table 3(b).

Refer to the distance matrix built in Phase II in the following steps to get the conflict between any two given blocks.

(1) Get an exam day D_j
(2) Get an exam block P_i within D_j
(3) Starting from P_i, build all possible paths not passing through any node twice. Besides, record the length of each path traversing it. Note that each path is identified by the source block number selected in(2).
(4) Go to (2) while $i \leq R$ (all blocks of D_j are exhausted)
(5) Choose the path with the minimum length to be the optimal arrangement of blocks in D_i
(6) Go to (1) while $j \leq D$ (all days are exhausted)

(a)

Refer to the distance matrix built in Phase II in the following steps to get the conflict between any two given blocks.

(1) Get an exam day D_i
(2) Starting from D_i, build all possible paths not passing through any node twice. Besides, record the length of each path traversing it. Note that each path is identified by the source day number selected in (1).
(3) Go to (1) while $i \leq D$ (all days are exhausted)
(4) Choose the path with the minimum length to be the optimal arrangement of the days.

(b)

Table 3.(a) Phase III, Part 1 Algorithm. (b) Phase III, Part 2 Algorithm.

4. Assigning Exams to Classrooms (*Phase IV*)

Phase III has prepared the actual schedule indicating the days, periods within days, and the courses assigned to them. Phase IV will be responsible for assigning the sections of these courses among the available classrooms during each period. Classrooms will be assigned one or more sections. Moreover, a section might be assigned to more than one classroom. The splitting mechanism should yield a maximum space utilization. It should be noted here that in order to be able to assign all of the final exams in a given period to classrooms, the total seating capacity of all the classrooms must be $\geq$ RC times the total number of students in that period. RC is the ratio of classroom capacity to the number of students to be assigned to it, for instance, in some institutions they assign to a classroom whose capacity is T, T/2 students, then RC = 2 in this case. Having this condition justifies the placing of an upper limit on the number of final exams assigned to the same block.

4.1 Formulation

The problem of assigning the final exams (scheduled in a given period) to available classrooms to minimize the number of split exams can be formulated as a non-linear integer program. This formulation is presented in table 4.

Let $q_{ij} = 1$ if exam i is assigned to room j and 0 otherwise.
Let $p_{ij} = 1$ be the number of students from exam i which are to be assigned to room j.

$$\min v = \sum^{Nt}_{i=1} \sum^{R}_{j=1} q_{ij} - N_t$$

subject to

1. $\sum^{Nt}_{i=1} q_{ij} <= 1$ for j = 1,2,...,R
2. $2p_{ij} <= q_{ij}C_j$ for i = 1,2,..., N_t for j = 1,2,...,R
3. $\sum^{R}_{j=1} p_{ij} = E_i$ for i = 1,2,...,N_t

$q_{ij} = 0,1$, $p_{ij} \in \{0,1,...,E_i$ for all i and j

Where N_t = number of exam scheduled in period t;
R = number of available classrooms;
C_j= the seating capacity of classroom j
E_i = the number of students in exam i

Table 4. Non-linear integer programming formulation of phase IV.

In the above formulation, v represents the number of rooms required beyond the number of exams (because of splitting) and is to be minimized. Constraint (1) forces each room to be assigned to at most one exam. Note that such a constraint is not applied in FESP. Constraint (2) ensures that the seating capacity is twice the number of students scheduled into that room. Note that in FESP the 'twice' factor is a variable set before running FESP, consequently, this variable might have the values of 1, 1.5, etc. Also, constraint (2) sets the binary variable q_{ij} to 1 if exam i is to be assigned to room j. Constraint (3) ensures that all participants of exam i will be assigned. Such a formulation is a non-linear integer

programming problem with N_t * R continuous variables, N_t * R binary variables, and $2N_t$ * R + N_t + R constraints. The size of this problem is too large for a practical application.

4.2 Phase IV Heuristic Procedure

Let $v_1, v_2, ..., v_n$ be the assigned exams to period t. Let the available rooms be $r_1, r_2, ..., r_p$. Let E_{vi} be the number of students participating in exam v_i and C_{rj} be the seating capacity of room r_j. The steps of the heuristic are as follows:

Step-0:

1. Sort all the *given* exams in descending order of number of students
2. Sort *all* the classroom in descending order of seating capacity.
3. Let v[1], v[2], ..., v[n] be the ordered exams, r[1], r[2], ..., r[p] be the ordered classrooms.
4. Set the variable $a_k = 0$ where k = 1,2, ..., p, (value of 0 for a_k indicates that room k has not been assigned yet), i = 1, j = 1, and move to **Step-1**.

Step-1:

If $a_j = 0$(Room j is not yet assigned), Move to **Step-2**. Increment j. If j > p the problem is unfeasible. Otherwise repeat **Step-1**.

Step-2:

If $E_{v[i]} > \lceil 1/2\ C_{r[j]} \rceil$, move to **Step-3** (i.e. the room capacity can not fit $E_{v[i]}$). Otherwise, assign exam v[i] to room r[j]. Let $a_j = 1$, increment i and j. If i > n, stop, all exams have been assigned. Otherwise move to **Step-1**.

Step-3:

Let $\delta = E_{v[i]} - \lceil 1/2\ C_{r[j]} \rceil$. Assign δ students of exam v[i] to room r[j], let k = p, q = 0 and move to **Step-4**.

Step-4:

If $k \leq j$, move to **Step-5**. If $a_k = 1$, decrease k and repeat **Step-4**, Otherwise, goto **Step-6**.

Step-5:

If q = 0, stop. There is no feasible solution. Otherwise assign δ students of exam v[i] to room r[q]. Let $a_q = 1$, $\delta = \delta - \lceil 1/2\ C_{r[j]} \rceil$, k = p, q= 0 and move to **Step-4**

Step-6:

If $\delta > \lceil 1/2\ C_{r[j]} \rceil$, move to **Step-7**. Otherwise assign the remaining students of exam v[i] to room r[k]. Let $a_k = 1$, increase i and j and move to **Step-1**.

Step-7:

Let q = k, k = k - 1, and move to **Step-4**.

CHAPTER 4

DESIGN AND IMPLEMENTATION

This chapter presents the design strategy and the implementation techniques we used in the development of **FESP**. The system is structured into a preparation sub-system and four modules which stand for the phases. In this chapter, each will be described in terms of the input it requires, the output it produces, the way it is functioning, and the data structures it uses. The **FESP** system chart shows these phases in figure 4.1.

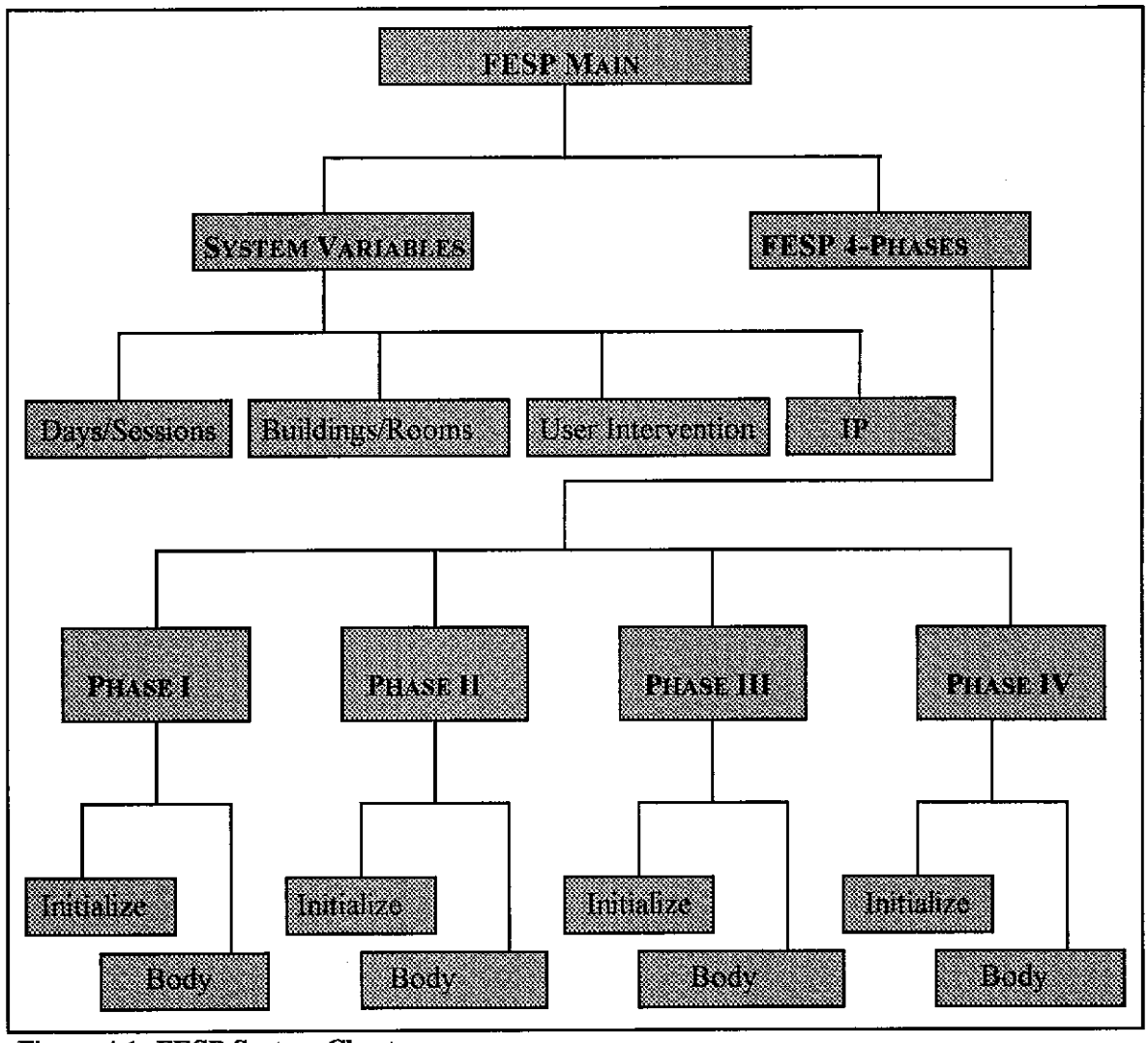


Figure 4.1: FESP System Chart

1. System Variables:

FESP system variables module is subdivided into four parts, days/periods part, buildings/rooms part, the user intervention part, and the system interface part (INTERFACE PROGRAM).

1.1 Days/Periods

In this part the user specifies the number of exam days (**ED**), and the number of exam sessions within a day (**ES**). The number of sessions per day is the same for all days. Automatically, the system numbers these days, *i.e.* if **ED** is 5, a list of five records will be displayed. The user sets the reading days (**RD**) out of these records (newly introduced by **FESP**). **ES * (ED - RD)** determines the total number of available exam sessions. Then a list of records corresponding to the **ES** are listed for the user to assign a time period for each. The data structures used in this part are days (Figure 4.2) and block_status (Figure 4.3) and time_period (Figure 4.20).

days	This table holds a set of records each associated with an exam day.

Field Name	Field Type	Description	Level Affected by
DayNum	Number	Holds the serial code of days to be scheduled (temporary number during the phases)	Initialize-Phase II
DayDate	Date	Holds the date of the exam day	System Variables

Figure 4.2: Days Structure Description.

Field Name	Field Type	Description	Level Affected by
ActualDayNum	Number	Holds the day serial code. This is the actual position of the day in the examination period. For reading or fixed days, this value will not be changed after scheduling.	Phase III Output
Reading	Char(1)	'Y' if this day is a reading day, otherwise, 'N'.	System Variables
Fixed	Char(1)	'Y' if this day is to be fixed in the ActualDayNum after the scheduling, otherwise, 'N'.	User Intervention

Figure 4.2: Days Structure Description.

Block_status	This table holds a set of records associated with each session irrespective of exam days, i.e. it holds the total number of exam sessions (called blocks) that are available for scheduling and equal to ES * (ED - RD).

Field Name	Field Type	Description	Level Affected by
BlockNum	Number	Holds the block serial code.	System Variables automatic generated
Status	Char(1)	Holds the status of the block a) available: More exams can be assigned to this session. b) full: No more exams can be assigned to this session.	Phase I.
UpperLimit	Number	Holds the maximum number of exams that can be assigned to the session.	System variables-automatic generated.
OverLoaded	Char(1)	Whether the given session was overloaded with assigned exams	Phase I.
ActualNum	Number	The final block number with a given day	User Intervention
Fixed	Char(1)	Whether the block is fixed or not in the actual number	User Intervention
DayNum	Number	The day number to which this block will be assigned	Phase II

Figure 4.3: Block_Status Structure Description.

1.2 Buildings/Rooms

In this part the user specifies which buildings/rooms are available during the examination period. A list of buildings/rooms is displayed and the user flags which ones are to be used for scheduling. The user also specifies the ratio of room seating capacity to the number of students to be assigned to it (**SR**). The tables used in this part are described in Figures 4.4 and 4.5.

Building	This table holds the buildings to be used in the scheduling process.

Field Name	Field Type	Description	Level Affected by
Code	Number	Holds the building serial code	System Variables IP
Name	Char(15)	Building name	System Variables IP

Figure 4.4: Building Structure Description.

Room	This table holds a set of records each associated with a room number

Field Name	Field Type	Description	Level Affected by
RoomNum	Number	Holds the room serial code.	System Variables IP
Building	Number	Holds the building code to which this room belongs	System Variables IP.
Assigned	Char(1)	Whether there exist any free space in this room to be assigned	Phase IV
Capacity	Number	Seating capacity of the room	System Variables IP
AssignCapacity	Number	How many students can be assigned to this room during the same exam (Capacity/SR)	System Variables and Phase IV.
Exam	Char(1)	If this room is to be used for examination	System Variables

Figure 4.5: Room Structure Description.

1.3 User Intervention

In this part the user sets a pre-defined sub-schedule for some courses. This pre-defined sub-schedule will not be affected during the scheduling process rather it will affect the scheduling process. Each sub-schedule will represent a group of courses/ections each assigned to an exam period, to an exam day, and to a classroom . The user might not specify a classroom, but he/she must assign an exam period (session) and an exam day. Finally, the user should specify one of these groups as being active during the current run of **FESP**. Another group might be chosen with a new run, and so on. Analysis might be done to choose the group that yields the best results. The user intervention issue is newly introduced by **FESP**. The data structures used in this part are Force_Schedule (Figure F.1) and Force_Groups (Figure F.2)

Force_Schedule	**This table holds some pre-defined schedule.**		
Field Name	**Field Type**	**Description**	**Level Affected by**
CourseNum	Char(10)	Holds the course number	User Intervention
SecNum	Number	Holds the section number	User Intervention
Enrollment	Number	Holds the enrollment in the given section that is assigned to the given room number.	User Intervention
DayNum	Number	Day assigned	User Intervention
BlockNum	Number	Session Assigned	User Intervention
RoomNum	Char(8)	Classroom assigned	User Intervention
Building	Number	Buiding number	User Intervention
Group	Number	Group serial number of the giveb sub-schedule	User Intervention

Figure F.1: Force_Schedule Structure Description.

Force_Groups	This table holds the code and the name of the differnt pre-defined schedules.

Field Name	Field Type	Description	Level Affected by
Code	Number	Group serial Number	User Intervention
Name	Char(12)	Group Name	User Intervention

Figure F.2: Force_Groups Structure Description.

1.4 Interface Program (IP)

The interface program (**IP**) links **FESP** to the database files of the computer system being run in the given institution. With respect to **LAU**, the computer system being run was developed under **ORACLE**, so the interface program is a very simple batch job that reads the information about students, rooms, buildings, and courses directly from the LAU computer system tables (date files) without the need for any data conversion. This program will be adjusted in accordance to the computer system on which **FESP** will be installed. Consequently, the host table names are to be identified to this program. This interface program is newly introduced by **FESP**. **IP** imports data to the following **FESP** tables: Building (Figure 4.4), Room (Figure 4.5), Registered (Figure 4.6), Course (Figure 4.7).

Registered	This table holds the IDs of the registered students and the courses/sections in which they are registered. This data is for a specific semester.

Field Name	Field Type	Description	Level Affected by
ID	Char (6)	Holds the student ID	System Variables IP
Course Number	Char (10)	Holds the course numbers	System Variables IP
Section Number	Char (2)	Holds the section numbers	System Variables IP

Figure 4.6: Registered Structure Description.

Course	This table holds the set of the courses to be scheduled.

Field Name	Field Type	Description	Level Affected by
Course Number	Char (10)	Holds the course numbers	System Variables **IP**
Course Title	Char (22)	Holds the course titles	System Variables **IP**
Credits	Number	Course Credits	System Variables **IP**

Figure 4.7: Course Structure Description.

2. Phase I:

Phase I is subdivided into two parts, the *initialization* and the *body* parts. The initialize part gets the data prepared by the interface program (**IP**) and prepares it for phase I body. The body of phase I prepares the input of phase II initialize part.

2.1. Phase I Initialize

2.1.1. Input

The input of the initialize part of phase I is a list of student schedules. Table 4.1 shows sample instances prepared by the **IP** as input to phase I.

Student ID	Course #	Section #
XY0001	CRS001	SEC001
XY0001	CRS005	SEC003
YZ0001	CRS001	SEC004
YZ0001	CRS006	SEC002
...	...	...

Table 4.1: IP output sheet, Initialize-phase I input sheet.

2.1.2. Output

The output of the initialize part of phase I is a list of courses sorted in ascending order by the weighted degree of each course. Table 4.2 shows sample input prepared by initialize-phase I.

Course #	W-Degree
CRS001	W100
CRS005	W900
CRS006	W800
...	...

Table 4.2: Initialize-phase I output sheet.

2.1.3. Implementation

Given:

- All of the final exams are represented as graph nodes (Figure 3.1).
- The weighted degree of a node is defined as: $w_i = d_i \sum a_{ik}$ (**WF**), where d_i is the degree of node i and $\sum a_{ik}$ is the total number of students taking exam i and all other exams in conflict with exam i.

1. Get the (next) final exam (f_i) using table (registered), if all exams are fetched then move to step **6**.

2. Compute T1 = number of enrolled students in f_i using table (registered).

3. Compute T2 = number of enrolled students in f_i and having other exams using table (registered).

4. Set W = T1 * T2 (using **WF**).

5. Insert the results, i.e. the course number and its weight, in table (fe_weight)

6. Assign each record in the fe_weight table a serial number to maintain the sorting order.

2.1.4. Structure Description

Initialize-phase I uses only two data structures. They are tables registered (Figure 4.6) and fe_weight (Figure 4.8).

Fe_weight	The data saved in this table represents an ordered list of courses.

Field Name	Field Type	Description	Level Affected by
Code	Number	A serial code that determines the position of each record in the ordered list.	FESP Initialize-phase I
Course #	Char (10)	The course numbers	FESP Initialize-phase I
WDegree	Number	The course weighted degree	FESP Initialize-phase I
BlockNum	Number	The Block number in which this course will be included.	FESP Phase I Body.
Enroll	Number	Enrollment.	FESP Initialize-Phase I

Figure 4.8: Fe_weight Structure Description.

2.2. Phase I Body

2.2.1. Input

The input of the phase I body consists of the following:

- The ordered list of courses with their weighted degree prepared in initialize-phase I (table 4.2). Such data is saved in table fe_weight (Figure 4.8).
- A list of available blocks (sessions) that are saved in table block_status (Figure 4.3). This sample list is prepared in table 4.3.

- Any pre-defined sub-schedules saved in table Force-Schedule (Figure F.1).

Block#	MaxLimit	OverLoaded	Status
B001	L	N	N
B002	L	N	N
B003	L	N	N
...	...	...	...

Table 4.3: Phase I body input sheet 2.

2.2.2. Output

The output of phase I body is a list of courses each assigned to a block number and saved in table fe_block_assign (Figure 4.9). Table 4.4 shows a sample list of this output.

Block#	Course#
B001	CRS001
B001	CRS004
B002	CRS002
...	...

Table 4.4: Phase I body output sheet.

2.2.3. Implementation

1. Set all blocks available, i.e. block_status.status = 'Y'.

2. If Any forced sub-schedule is active then perform **assign_forced.**

3. Fetch the first unassigned final exam (A) from the list prepared in initialize part (fe_weight). A final exam is said to be unassigned if there exist no records in table fe_block_assign (Figure 4.9) associated with this exam.

4. Find an available block (B) from table (block_status), such that if (A) is assigned to it, no conflicts occur between (A) and any other exam that was already assigned to (B). Here, we mean by conflicts:

a. Intersection of two exams condition (prerequisite condition either put on or off).

b. Exams given by the same instructor condition (either put on or off). In this implementation it is always off.

5. If (B) was found, then assign (A) to (B) and go to **3**. Otherwise, i.e. no block was found to satisfy the conflict conditions, choose the block (B1) which satisfies the following condition: If (A) was to be assigned to (B1), then the minimum conflict will occur. In this step some deviation from the heuristic took place.

Assign_Forced

1. Get a block B_i using table Block_Status.

2. Get any block B_j with its day D_j using table Force_Schedule.

3. Assign all courses assigned to B_j and D_j in table Force_Schedule B_i using table fe_block_assign.

4. Set the fields Fixed = 'Y' and ActualNum = B_j and daynum = D_i in table block_status where blocknum field value = B_i.

5. Move to **1**.

2.2.4. Structure Description

Phase I uses several tables which are: fe_weight (Figure 4.8), block_status (Figure 4.3), fe_block_assign (Figure 4.9), registered (Figure 4.6) and stu_crs_blk (Figure 4.10).

Fe_block_assign	This table determines which courses are assigned to which blocks.

Field Name	Field Type	Description	Level Affected by
Course Number	Char (10)	Holds the course numbers	Phase I
BlockNumber	Number	Holds the block number	Phase I
Enroll	Number	Enrollment	Phase I

Figure 4.9: Fe_block_assign Structure Description.

Stu_crs_blk	This temporary table holds for a given course the conflict degree with all the blocks.

Field Name	Field Type	Description	Level Affected by
BlockNumber	Number	Holds the block number	Phase I
ConflictDegree	Number	Holds the conflict degree	Phase I

Figure 4.10: Stu_crs_blk Structure Description.

3. Phase II:

Phase II is subdivided into two parts, the initialize part and the body. The initialize part assigns blocks to days sequentially. The body of phase II searches in the sequential assignment for any improvement, if any exists, a pair-wise exchange is done for a couple of blocks.

3.1. Phase II Initialize

3.1.1. Input

The input of the initialize part of phase II is simply the data found in tables days (Figure 4.2) and block_status (Figure 4.3).

3.1.2. Output

The output of the initialize part of phase II is a list of blocks assigned to days saved in table day_assign figure 4.11. Table 4.5 shows sample instances prepare by initialize-phase II.

BlockNum	Day#	Period
B001	D001	
B002	D001	
...	...	...
B003	D002	
...	...	

Table 4.5: Initialize-phase II output sheet.

3.1.3. Implementation

To begin, the initial solution is obtained by assigning the exam blocks (Figure 3.2) to exam days sequentially. That is, the first P/D (P: Total number of exam blocks, D: Total number of exam days) to the first day, the second P/D blocks to the second day and so on. Note that fixed blocks positions are maintained (Newly introduced by **FESP**). Moreover, using table days, the field value fixed is set to 'Y', actual_Day_Num is set the same value in table Block-Status for the given block number.

3.1.4. Structure Description

The tables used in this part are block_assign (Figure 4.3), days (Figure 4.2) and day_assign (Figure 4.11)

Day_Assign	This table determines which blocks are assigned to which days. This table resembles matrix Y defined in chapter III. If a record is found in this table then that is equivalent to a 1 in Y matrix, otherwise, that will be equivalent to a 0 in Y matrix.

Field Name	Field Type	Description	Level Affected by
DayNumber	Number	Holds the day number	Initialize Phase II and Phase II Body
BlockNumber	Number	Holds the block number	Initialize Phase II and Phase II Body
Period	Number	Holds the session number within the day to which the given block is assigned.	Initialize Phase III.

Figure 4.11: Day_Assign Structure Description.

3.2. Phase II Body

3.2.1. Input

The input of the phase II body consists of the data prepared in the initialize part of phase II. Such data is saved in table day_assign (table 4.5).

3.2.2. Output

The output of phase II body is the same as that of the initialize part, but the block assignment is supposed to be improved in this body part (Table 4.5).

3.2.3. Implementation

After obtaining an initial solution, we examine possible improvements through a pair-wise exchange of exam blocks. If no further improvement is possible, then phase II processing is done. Note that fixed blocks are not included in this process. The steps of pair-wise exchange are as follows:

1. Prepare the Exam-Block-Intersection Matrix (C) that shows the intersection between any couple of blocks using tables fe_block_assign (Figure 4.9) and registered (Figure 4.6). Matrix (C) is represented as table stu_in_blocks (Figure 4.12).

2. Prepare the Block-Expected-Assignment-Conflict Matrix (R) that shows the degree of conflict resulted from assigning(virtually) any block B_i to each of the exam days D_i (B_i intersection with itself is not counted). This step depends on the matrix (C) prepared n step 1. Matrix R is represented as table temp_matrix_R (Figure 4.13).

3. Fetch a new (for later reference) block B_i, and get the day D_j in which the least conflicts result if B_i was assigned to it. However D_j should not be the one D_i in which B_i is currently assigned to. We call D_j the potential day, D_i the current day, B_i the current block. This step depends on matrix (R) prepared in step 2. When all B_i are exhausted then go to step 5.

4. For each block B_j in D_j, do the following: (we call B_j the potential block)

a. From matrix (R):

Get the conflict degree of the current block in the current day (T1).

Get the conflict degree of the current block in the potential day (T2).

Get the conflict degree of the potential block in the potential day (T3).

Get the conflict degree of the potential block in the current day (T4).

b. Compute the following:

I1 = (T2 - T1) : Represents the decrease/increase in conflict degree of the current block B_i when assigned to the potential day D_j.

I2 = (T4 - T3) : Represents the decrease/increase in conflict degree of the potential block B_j when assigned to the current day D_i.

Now, we select the B_j with the minimum A = I1 + I2 if any. If A < 0, that is, an improvement is detected, then exchange B_j with B_i. That is, assign B_j to the current day and B_i to the potential day. Go to step **2** to re-compute matrix (R) since the assignment structure has been just changed. Otherwise, if A >= 0, no improvement was detected, then go to step **3** to get another B_i.

5. If any exchange took place during the last iteration, reset the list pointer of step 3-B_i-list to TOP, go to step **3**. Otherwise, phase II is done.

3.2.4. Structure Description

The tables used in this part are temp_matrix_R (Figure 4.13), day_assign (Figure 4.11), stu_in_blocks (Figure 4.12), and temp_w (Figure 4.14).

Stu_in_blocks	This table resembles matrix C defined in Chapter III. It holds the number of conflicts between any two given blocks

Field Name	Field Type	Description	Level Affected by
BlockNumber1	Number	Holds the first block #.	Phase II Body
BlockNumber2	Number	Holds the second block #.	Phase II Body
Students	Number	Holds the number of students (conflicts) who have exams assigned to the given two blocks.	Phase II Body.

Figure 4.12: Stu_in_blocks Structure Description.

Temp_matrix_R	This table resembles matrix R defined in Chapter III. It holds a block number, a day number, and the conflict if this block is to be assigned to this day.

Field Name	Field Type	Description	Level Affected by
BlockNumber	Number	Holds the block #.	Phase II Body
DayNumber	Number	Holds the day #.	Phase II Body
Students	Number	Holds the number of conflicts that will arise if BlockNum is assigned to the given dayNum.	Phase II Body.

Figure 4.13: Temp_matrix_R Structure Description.

4. Phase III:

In Phase II, the blocks were assigned to exam days. Phase III in its two parts will assign each block to an exam session and arrange the exam days in order to minimize students with consecutive exams. The initialize part of phase III arranges the blocks within each exam day to minimize the first type of back-to-back conflicts (within exam days conflicts). Phase III body arranges the exam days to minimize the second type of back-to-back conflicts (between exam days conflicts). Note that any selection from table Day_Assign excludes the reading days. Reading days are dealt with in phase III body. Phase III heuristic is newly introduced by **FESP**.

4.1. Phase III Initialize

4.1.1. Input

The input of the initialize part of phase III is simply the data found in table day_assign (Figure 4.11). Table 4.6 shows the input sheet: Examination days and the blocks assigned to each day.

Day#	Block #
D001	B001 B003 ...
D002	B002 B004 ...
...	...

Table 4.6: Initialize-phase III input sheet.

4.1.2. Output

The Phase II - Initialize part assigns each block to an examination session. This is done by filling field period in table day_assign with the value of the session. This value is a serial number unique for each day. The output will then look like the sheet in table 4.7.

Day#	Block #	Period
D001	B001 B003 B007 ...	2 3 1 ...
D002	B002 B004 ...	1 2 ...
...	...	...

Table 4.7: Initialize-phase III output sheet.

4.1.3. Implementation

Given:

- All the blocks are assigned to days in table day_assign.
- The degree of conflict between any two given blocks is saved in table stu_in_blocks (Figure 4.12) prepared in phase II.

1. Get the (next) exam day (ED_i) using table day_assign (distinct value), if all days are fetched then STOP.

2. Clear tables path_length (Figure 4.15)and day_paths (Figure 4.14), and variable path_len for later use.

3: Get the (next) P_i from the exam blocks assigned to day ED_i. If all blocks are fetched, go to step 8.

4. Set P_i as the head of a list (L_i) to be built later. This list is saved in table day_paths.

5. Get the block P_j from table stu_in_blocks with the minimum degree of conflict (**DC**) with P_i. P_j should be assigned to ED_i and should not be on L_i yet. If the position in the list corresponds to a pre-assigned block then fix it, otherwise use the fetched block. If all blocks are fetched, go to step 7.

6. Add P_j to L_i and increment the path_len by **DC**. Go to step 5.

7. Insert in table path_length the values of the L_i head block and its associated path_len value. Move to step 3.

8. Choose from table path_length a list-head block (P_M) with the minimum associated path_len. The list in table day_paths whose head is P_M represents the block optimal arrangement within D_i. Update accordingly the field period in table day_assign. Move to Step 1.

4.1.4. Structure Description

The tables used in this part are day_assign (Figure 4.11), stu_in_blocks (Figure 4.12), day_paths (Figure 4.14), and path_length (Figure 4.15). The last two tables are used also in phase III for another purpose since they are just temporary tables.

Day_Paths	This table is used in the initialize part of phase III to hold the lists of blocks being built during the processing per day. While in the body part of phase III, it holds the lists of days being built during the processing.

Field Name	Field Type	Description	Level Affected by
DayCode	Number	Holds the serial number of block/day with the list	Initialize Phase III / Phase III Body
Day	Number	Holds the block/Day number	Initialize Phase III / Phase III Body
Source	Number	Holds the code of the head block/day of the list.	Initialize Phase III / Phase III Body

Figure 4.14: Day_Paths Structure Description.

Path_Length	This table is used in the initialize part of phase III to hold the head block of a given list with the associated list path length. While in the body part of phase III, it holds the head day of a given list with the associated list path length.

Field Name	Field Type	Description	Level Affected by
Source	Number	Holds the block/day code	Initialize Phase III / Phase III Body
Len	Number	Holds the path length of the list whose head is Source.	Initialize Phase III / Phase III Body

Figure 4.15: Path_Length Structure Description.

4.2. Phase III Body

4.2.1. Input

The input of the phase III body consists of the data prepared in the initialize part of phase III. Such data is saved in table day_assign (Table 4.5) and Table 4.7 represents that input.

4.2.2. Output

The output of phase III body is the same as that of the initialize part, but the days are assigned the actual day numbers, *i.e.,* the days are now numbered as they will be actually during the examination period. Moreover, each day will be assigned its date (the date of the first examination day was entered in the system variables part).

4.2.3. Implementation

Given:

- All of the blocks are assigned to days in table day_assign.
- All of the blocks are assigned to periods within days in table day_assign.
- The degree of conflict between any two given blocks is saved in table stu_in_blocks (Figure 4.12) prepared in phase II.

Part A:

1. Clear table Day_Distance_Matrix (Figure 4.16) for later use.

2. Get the (next) exam day ED_i using table day_assign (distinct value), if all days are fetched then perform **Part B.**

3. Get the (next) exam day ED_j which is different from ED_i using table day_assign (distinct value), if all days are fetched then move to step 2.

4. Get the block P_L assigned in the last session in ED_i, and the block P_F assigned in the first session of ED_j.

5. Get the conflict between P_L and P_F using table stu_in_blocks (Figure 4.12) and save this information in table Day_Distance_Matrix. Move to step 3.

Part B:

1. Clear tables path_length (Figure 4.15) and day_paths (Figure 4.14), and variable path_len for later use.

2. Get the (next) exam day ED_i using table day_assign (distinct value), if all days are fetched then move to step 8.

3. Set ED_i as the head of a list L_i to be built later. This list is saved in table day_paths.

4. If the next available position in list L_i is set to be a reading or fixed pre-assigned day (using table days), then fix it as should be and continue. This checking is done later for each available list position.

5. Get the day ED_j using table Day_Distance_Matrix prepared in **Part A** with the minimum degree of conflict (**DC**) with ED_i. ED_j should be not be on L_i yet. If all days are fetched then move to step 7.

6. Add ED_j to **L** and increment the path_len by the **DC**. Repeat step **5**.

7. Insert in table path_length the values of the L_i head day and its associated path_len value. Move to step **3**.

8. Choose from table path_length a list-head day (D_M) with the minimum associated path_len. The list in table day_paths whose head is D_M represents the days optimal arrangement within the examination period. Update accordingly the field actual_day_num and daydate in table days (Figure 4.2). STOP.

4.2.4. Structure Description

The tables used in this part are day_assign (Figure 4.11), stu_in_blocks (Figure 4.12), day_path (Figure 4.14), path_length (Figure 4.15), and Day_Distance_Matrix (Figure 4.16).

Day_Distance_Matrix	**This table holds the degree of conflict between the block assigned to the last session of a given day and the block assigned to the first session of the next day.**

Field Name	Field Type	Description	Level Affected by
Day1	Number	Holds the first day#.	Phase III Body
Day2	Number	Holds the second day#.	Phase III Body
Conflict	Number	Holds the degree of conflict between Day1 and Day2.	Phase III Body.

Figure 4.16: Day_Distance_Matrix Structure Description.

5. Phase IV:

In Phase III, the final exam schedule is completed partially and what remains for phase IV to do is just to specify the place (classroom and building) for each examination session. Phase IV in its two parts will assign for each course section final exam a place which consists of one or more classrooms in the same or different buildings based on the seating capacity of the available classrooms. The initialize part of phase IV prepares a list that contains the course numbers, the section numbers, the enrollment in each section, the assigned exam day and session.. Phase IV body assigns classrooms for each of these sections.

5.1. Phase IV Initialize

Phase IV initialize part job is a very simple read-write process. Information are read from tables registered (Figure 4.6), block_status (Figure 4.), day_assign (Figure 4.11), and fe_Block_Assign (Figure 4.9). This information is then written into table schedule (Figure 4.17).

5.1.1. Output

Phase IV - Initialize part will fill table schedule as mentioned above. The output will look like table 4.8

5.1.2. Structure Description

The tables used in this part are registered (Figure 4.6), block_status (Figure 4.), day_assign (Figure 4.11), fe_Block_Assign (Figure 4.9) and schedule (Figure 4.17).

Course #	Section #	Enrollment	Assigned Day#	Assigned Session
CS201	A	24	D001	1
CS201	B	25	D001	1
MT101	A	18	D002	3
BI211	A	20	D004	2
...	...	...	...	...

Table 4.8: Initialize-phase IV output sheet.

Schedule	This table holds the final exam schedule (except place assignment).

Field Name	Field Type	Description	Level Affected by
CourseNum	Char(10)	Holds the course number	Initialize Phase IV
SecNum	Number	Holds the section number	Initialize Phase VI
Enrollment	Number	Holds the enrollment in the given section.	Initialize Phase VI
Sched_Enrollment	Number	Number of students in this section to be assigned a room	Phase IV Body
AssignedDate	Number	Day assigned	Initialize Phase IV
AssignedPeriod	Number	Session Assigned	Initialize Phase IV

Figure 4.17: Schedule Structure Description.

5.2. Phase IV Body

5.2.1. Input

The input of the phase IV body consists of the data prepared in the initialize part of phase III (table 4.8). Classrooms information saved in table room (Figure 4.5). Tables day_assign (Figure 4.11) and fe_block_assign (Figure 4.9) are used also for reference purposes.

5.2.2. Output

The output of phase IV body is the Final Examination Schedule that is saved in two tables: Schedule (Figure 4.17) and Room_Schedule (Figure 4.18).

5.2.3. Implementation

Given:

- The ratio (**RR**) of the room seating capacity over the number of students to be assigned to each room using table sched_var (Figure 4.19).
- Each section is assigned to a day/session.

Preparation Step:

- Use table room to set the value of AssignCapacity (**AS**) = Room capacity / **RR**.
- Update table schedule set the value of Sched_Enrollment = Enrollment.
- In case there is any available pre-assigned sub-schedules perform **assign_force.**

1. Clear table Room_Schedule (Figure 4.19) for later use.

2. Get the (next) block P_i using table day_assign, if all blocks are fetched then **Stop.**

3. Get the first available classroom R_i of building B_i with the maximum **AS** using table room (Figure 4.5), if all rooms are fetched then move to step **9**.

4. Get the first unassigned course number (C_i), section number (S_i) and its enrollment (E_i) using table schedule. C_i should belong to P_i using table fe_Block_Assign figure (4.9). If all S_i of C_i are assigned to classrooms then move to step **2**.

5. If classroom **AS** >= E_i then perform **assign_section** with parameters C_i, S_i, Ri, B_i, **AS** and E_i, and move then to step **3**. Otherwise, continue.

6. Perform **assign_section** with parameters C_i, S_i, Ri, B_i, **AS** and **AS**.

7. Get the first available classroom R_j of building B_j with the minimum **AS** using table room (Figure 4.5), if all rooms are fetched then move to step **9**.

8. Move to step **5**.

9. If there exist any section in block P_i which is not yet assigned, then there is an unfeasible room assignment solution for P_i sections. Otherwise Move to step **2**.

Assign_Force

For each pre-assigned course number/section number perform **assign_section** using table Force_Schedule.

Assign_Section Parameters (Course, Section, Room, Building, RoomCap, NumToAssign)

1. Insert into table Room_Schedule figure (4.18) the parameter values.

2. If NumToAssign = RoomCap then set R_i as non-available room (table room field assign gets 'Y'), otherwise, set AssignCapacity in table room to RoomCap - NumToAssign.

3. Set field Sched_Enrollment in table Schedule = Sched_Enrollment - NumToAssign. If Sched_Enrollment becomes 0, then this means that S_i is assigned.

5.2.4. Structure Description

The only table affected in this part is Room_Schedule (Figure 4.18).

Room_Schedule	This table holds the assignment of building/classrooms to final exam sections.

Field Name	Field Type	Description	Level Affected by
Course Number	Number	Holds the Course Number	Phase IV Body
Section Number	Number	Holds the Section Number	Phase IV Body
Assigned Room	Number	Holds the Room Number	Phase IV Body.
Assigned Bldg.	Number	Holds the building Number	Phase IV Body.
Total Assigned	Number	How many students of the given section were assigned to the given room.	Phase IV Body.

Figure 4.18: Room_Schedule Structure Description.

Sched_Var	This table holds FESP system parameters.

Field Name	Field Type	Description	Level Affected by
Today	Date	Holds the date of the first exam date.	System Parameter to **FESP**
BlkPerDay	Number	Holds the number of exam sessions with each exam day.	System Parameter to **FESP**
RoomCapRatio	Number	Holds the ratio of the classroom seating capacity over the number of the to-be-assigned students	System Parameter to **FESP**
PreRequiste	Char(1)	To set the conditions of pre-requisite on or off.	System Parameter to **FESP**
BlockLimit	Number	Students Limit per Session	System Parameter to **FESP**

Figure 4.19: Sched_Var Structure Description.

Period_Time	This table holds time-text strings for each exam session.

Field Name	Field Type	Description	Level Affected by
Period	Number	The session number within an exam day.	System Parameter to **FESP**
Text	Char(11)	Time string associated with the given session	User Defined, System Variables

Figure 4.20: Period_Time Structure Description.

5. Summary:

In this chapter we presented the implementation details of the different **FESP** parts. However, we did not mention any thing about the structure of the reports generated by **FESP** mainly the Schedule and the analysis report. This is because they are just a read-only processes that take their data from the above described tables.

In this version of **FESP** we have followed a database approach in the analysis, design and even in the implementation phases. Using databases in such tools might yield an increase in the execution time on one hand, but on the other hand it provides a flexible means for any customization or maintenance. Chapter 5 will show the experimental results and present an analysis on the execution aspects of **FESP**.

CHAPTER 5

EXPERIMENTAL RESULTS

We present in this chapter the results of two experiments done using **FESP**. The first experiment is done on a real-life set of data taken from **LAU** records for the Spring 1994-1995 semester using the same parameter values set in the manual scheduling process. In the second experiment we have changed one of these parameters (the number of exam days) to show the sensitivity of **FESP** to this parameter and how the output will be affected in the sense of conflicts minimization. Comparative results done between **FESP** output schedule and the manually prepared schedule at **LAU** is presented in several graphs.

5.1 Real-Life Data Experiment 1

FESP needs several parameters to be set a priori, for example, the number of exam days, the number of available rooms, the number of sessions within exam days, etc. Moreover, the data, mainly the student schedules, should also be prepared. In this experiment we have used the same parameters' values used in the manual preparation process and the same set of student schedules. The main parameters and other attributes values set before the experiment are shown in table 5.1 below.

First, the manually prepared schedule was entered into the **ORACLE** database. Then it was imported into **FESP** pre-assigned schedule table, i.e. we have considered that all the sections found in this schedule are pre-assigned. **FESP** process and the analysis procedure ran over this pre-assigned schedule and produced the results shown in Figures 5.1, 5.2.

Attribute	Value	FESP/Manual
Total Number of registered Students	2664	Common
Total Number of Offered Courses	227	Common
Total number of Students-Exams	9887	Common
Total Number of exam days	6	Common
Total Number of Reading Days	3	Common
Number of Sessions within each Exam Day	4	Common
Number of Available Classrooms	21	Common
Total Seating Capacity of Classrooms	888	**FESP**
Session Upper Limit	750	**FESP**
Number of enforced-sections	0	**FESP**

Table 5.1: Input Parameters to the real-life data experiment.

Second, **FESP** process ran over the same sample data but with no pre-assigned sections. The results obtained after this run are show in Figures 5.1, 5.2. In Figure 5.3 a time comparison is presented between the two schedules in terms of hours. The results show that the solution quality obtained by **FESP** is much superior from that obtained by LAU schedule. In Table 5.2 the results are presented showing the three conflict measures:

1. 1st order conflict: Students with simultaneous exams.

2. 2nd order conflict A: Students with more than two exams per day.

3. 2nd order conflict B: Students with more than three exams per day.

4. 3rd order conflict: Students with consecutive exams.

	1st Order	2nd Order A	2nd Order B	3rd Order
FESP	0	48	1	751
Manual	44	114	2	1077

Table 5.2: Comparative Results between FESP and LAU Manual Schedule

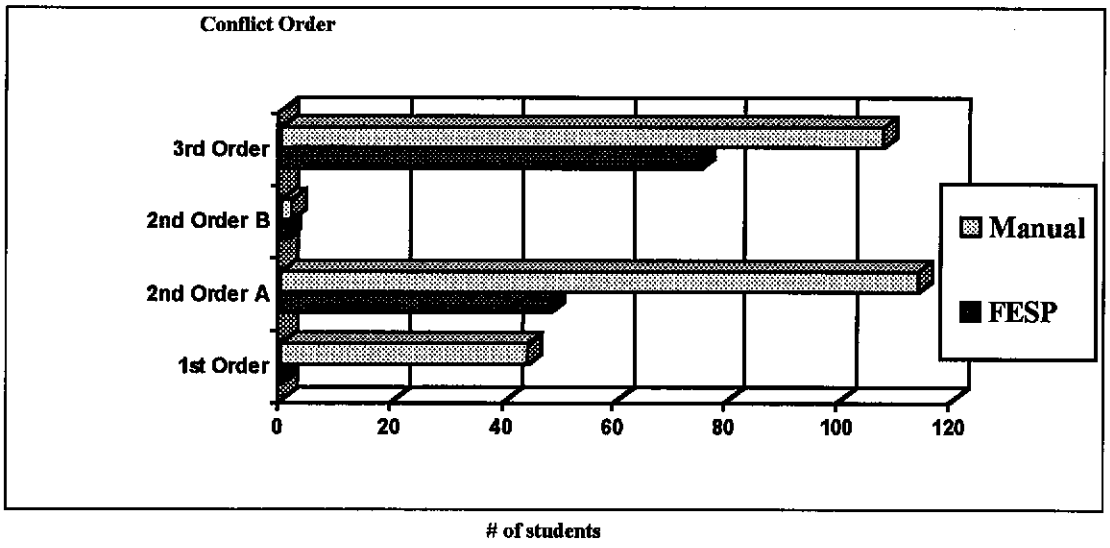


Figure 5.1: Comparative Graph 1 (FESP Versus LAU Manual Schedule)

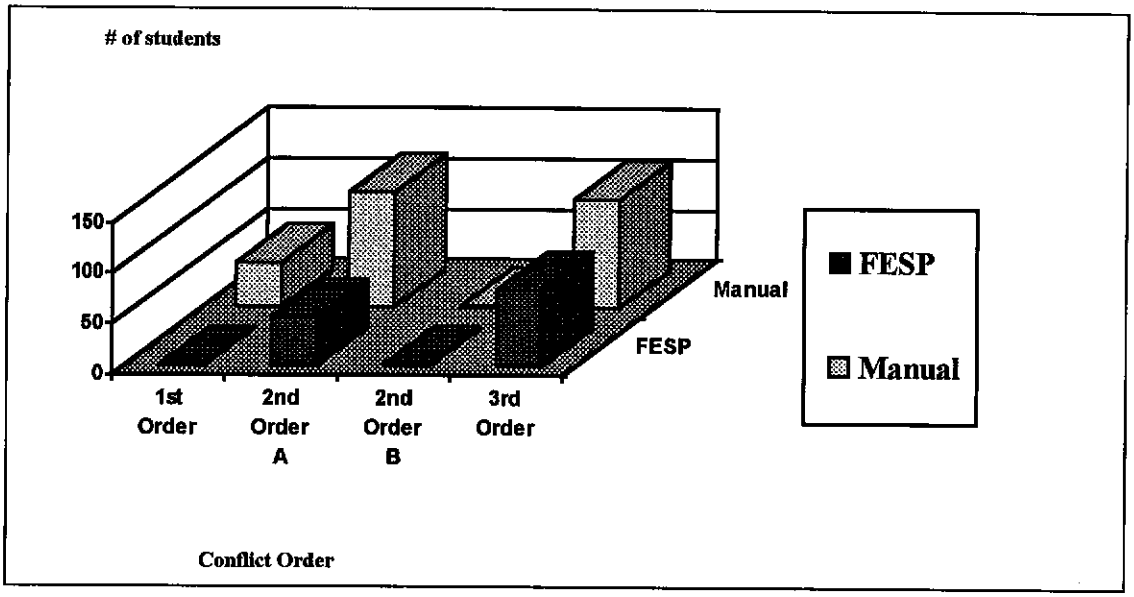
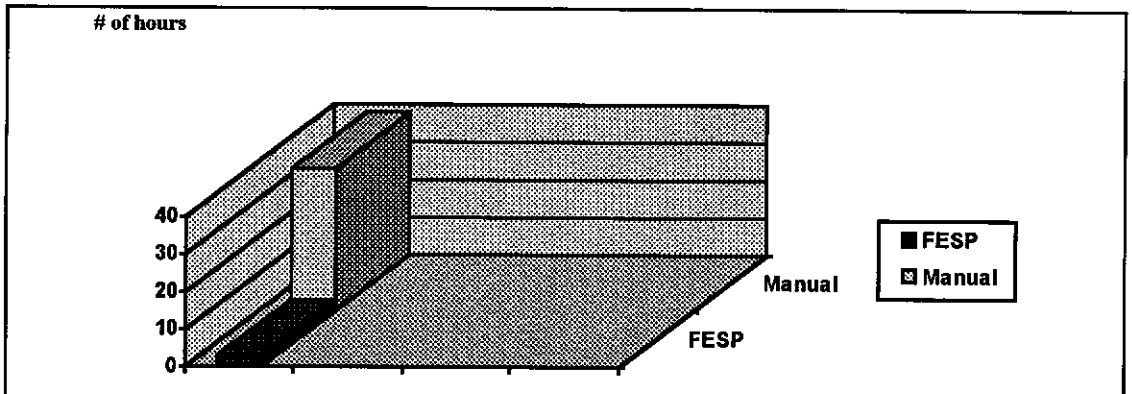


Figure 5.2: Comparative Graph 2 (FESP Versus LAU Manual Schedule)



The scale used in this graph is execution time in hours (or total preparation time since in the manual work there is no real execution time). This graph shows how much superior **FESP** is (3 hours) over the manual work (38 hours without the student schedules checking that might take months).

Figure 5.3: Time Comparison (FESP Versus LAU Schedule)

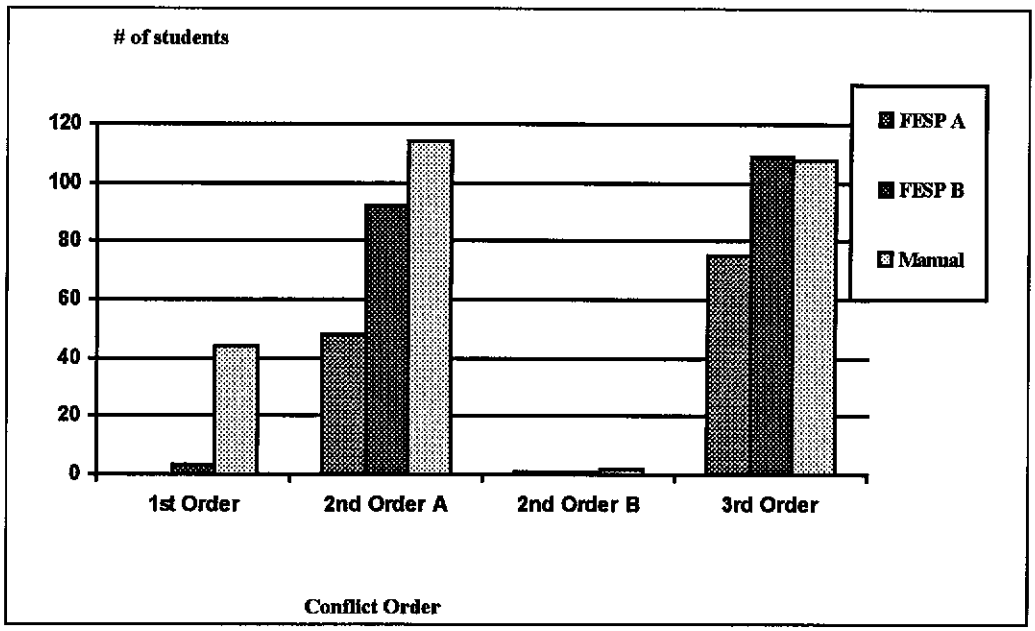


Figure 5.4: Comparative Graph 4 (FESP A, FESP B, LAU Schedule)

CHAPTER 6

RELATED WORK

Most educational institutions face the problem of scheduling exams at the end of each semester. Typically, the most important objectives in this problem are the minimization of (1) the number of students with simultaneous exams, (2) the number of students with more than two exams per day, (3) the number of students with back-to-back exams, and the (4) maximization of the space utilization. For large-scale problems involving thousands of students and exams, the size of the problem forces it to be decomposed into distinct phases so as to make the solution process manageable. Several good attempts were done in each of these issues. In this chapter, we try to highlight some of these attempts.

One of the attempts in this direction was the introducing of a computer-assisted multistage participative procedure for examination scheduling [Romero 1982]. The requirements were categorized into three groups. The Administration, departments, and the students requirements. Then the problem was divided into three stages. In stage 1, negotiation is done between the representatives of the three groups. In stage 2, examination dates are settled, and checking for student / faculty conflicts is re-assured. In the last stage, the assignment of classrooms will be done. A flow chart is prepared for this process [Romero 1983, Figure 3] and several programs were developed to automate certain tasks within

each stage e.g. setting the examination dates and reading student information from their files. As it is clear, the major part of the process was done manually.

Mehta tried to tackle the problem by the application of a Graph Coloring Method [Mehta 1991]. The problem was formulated in graph coloring, each course examination is represented by a vertex. An undirected edge is introduced between any two vertices if the examinations represented by them have conflicts. To solve the problem is equivalent to find a vertex coloring of the graph using the smallest number of colors such that the two vertices which are joined by an edge are not assigned the same color. Each color corresponds to a time frame. The application is divided into three stages. In stage 1, the requirements are translated into the graph e.g. if two courses are to be taken simultaneously, then they share the same graph vertex. In stage 2, an attempt to combine certain time frames is made in order to fit the examination period. In the last stage, time frames are rearranged in order to minimize number of consecutive exams, however, the day-to-day type of consecutive exams is not considered. When the number of students and courses to be scheduled is small, it may be possible to reach an efficient solution by using such methods as the Graph Coloring used in this application [Balakrishnan et al. 1992]. For instance, in this application, the number of students was only 750 enrolled in 84 courses.

Balakrishnan, Lucena, and R.T. Wong presented a scheduling method to reduce the back-to-back type conflict [Balakrishnan et al. 1992]. The exam groups were assumed to be such that the number of students required to take two or more exams in the same group is minimum. The time-slots were such that the last slot of any day and the first slot of the following day are not considered back-to-back. They suggested building a network structure that assumes each exam to be assigned to each period in all days. Then the problem turned out to be a time-dependent TSP (Travel Salesman Problem) which is considerably more difficult to solve than the TSP [Lucena 1990]. By computing certain lower and upper bounds to the objective value of the problem, and then applying Lagrangian Relaxation to improve these bounds, the weights of network (Complete Directed Graph) arcs will be updated until an acceptable path (solution) is achieved. They claimed (based on their tests) that their proposed procedure

generates a near-optimal feasible schedules for realistically-sized exam scheduling problems.

CHAPTER 7

FURTHER WORK AND CONCLUSION

We have developed a **Final Exam Scheduling Package** which minimizes the number of exam conflicts and maximizes space utilization. First, second and third order conflicts i.e. Students with simultaneous exams, students with more than two exams per day, and students with consecutive exams were minimized. Splitting sections into more than one room was minimized and sharing the same room among exams was maximized. This tool can be used at **LAU** or other universities with minor customization. The experimental results showed that **FESP** yields much better results than those obtained by the manually prepared exam schedule being used at **LAU**.

FESP can be enhanced in a number of ways to make it more general and more flexible. The specifications for such enhancements are as follows:

1. The ability of having variable number of exam sessions within each exam day i.e. being able to assign for each exam day a different number of exam sessions. This will allow also having variable-time sessions (2-hour sessions, 3-hour sessions, etc.).

2. The ability of having user intervention with more features than what currently exist. For example: (a) the ability to pre-assign final exams only to days without specifying the sessions, or only to sessions without specifying the days. (b) the ability to enforce the scheduling of a group of exams together within the same given session or the same given day without specifying the actual session or day, or the contrary. (c) the ability to interfere in the schedule after its production in order to perform the analysis procedure on the given changes without the need of schedule re-production.

3. Adding the Instructor's Constraint to the set of constraints considered in the grouping of exams process, i.e. preventing exams of the same instructor to be scheduled simultaneously.

4. The ability of scheduling sections of the same course in different sessions.

5. Finally, minimizing the execution time will always be an uncompleted job that needs more improvement.

Appendix A

Final Exams Schedule (FESP Work)
SPRING 94-95

Saturday 10, Jun 1995

Time	Course #	Sec.	Course Title	Place
8:00-10:00	2223	A	PSYCH.OF YOUNG CHILD	LRC21
	2223	B	PSYCH.OF YOUNG CHILD	Sg.12

Time	Course #	Sec.	Course Title	Place
10:30-12:30	1232	A	ORGANIC CHEMISTRY II	LRC21
	1735	A	INTROD.COMP.SYSTEMS	Sg.12
	3336	A	MODERN ENG.GRAMMAR	N.215
	5511-2	A	ENGLISH II	Sg.6
	5511-2	B	ENGLISH II	N.GR.FL
	5511-2	C	ENGLISH II	Sg.27
	5511-2	D	ENGLISH II	N.208
	5511-2	E	ENGLISH II	N.GR.F.3
	5511-2	F	ENGLISH II	LRC21
	5511-2	G	ENGLISH II	N.GR.F.4
	5511-2	H	ENGLISH II	Sg.12
	5511-2	I	ENGLISH II	N.213
	5521	A	COMMUNICATION ART	N.GR.F.2
	5521	B	COMMUNICATION ART	O.G.B.
	5521	C	COMMUNICATION ART	N.113
	5521	D	COMMUNICATION ART	O.G.B.
	5521	E	COMMUNICATION ART	LRC21
	5521	F	COMMUNICATION ART	Sg.10
	5521	G	COMMUNICATION ART	N.206
	5732	A	HISTORY OF LEBANON	Sg.29

Time	Course #	Sec.	Course Title	Place
1:00-3:00	4826	A	BUSINESS LAW	LRC21
	4826	B	BUSINESS LAW	O.G.B.
	4826	C	BUSINESS LAW	Sg.12

Time	Course #	Sec.	Course Title	Place
3:30-5:30	5510	A	ENGLISH I	Sg.12
	5510	B	ENGLISH I	LRC21
	5510	C	ENGLISH I	Sg.12
	5510	D	ENGLISH I	N.GR.F.3
	5510	E	ENGLISH I	O.G.B.
	5510	F	ENGLISH I	Sg.10
	5510	G	ENGLISH I	N.GR.FL
	5510	H	ENGLISH I	O.G.B.
	5512-1	A	ENGLISH III	Sg.27
	5512-1	B	ENGLISH III	Sg.29
	5512-1	C	ENGLISH III	O.G.B.
	5512-1	D	ENGLISH III	LRC21
	5512-1	E	ENGLISH III	N.GR.FL
	5512-1	F	ENGLISH III	N.GR.F.4
	5512-1	G	ENGLISH III	LRC21
	5512-1	H	ENGLISH III	N.213

Final Exams Schedule (FESP Work)
SPRING 94-95

Saturday 10, Jun 1995

Course #	Sec.	Course Title	Place
5512-1	I	ENGLISH III	N.206
5522	A	SOPHOMORE RHETORIC	N.215
5522	B	SOPHOMORE RHETORIC	LRC21
5522	C	SOPHOMORE RHETORIC	N.206
5522	D	SOPHOMORE RHETORIC	N.GR.F.2
5522	E	SOPHOMORE RHETORIC	N.GR.F.1
5522	F	SOPHOMORE RHETORIC	N.208
5522	G	SOPHOMORE RHETORIC	N.113
5522	H	SOPHOMORE RHETORIC	Sg.6
5522	I	SOPHOMORE RHETORIC	Sg.10
5522	J	SOPHOMORE RHETORIC	Sg.12

Final Exams Schedule (FESP Work)
SPRING 94-95

Monday 12, Jun 1995

Time	Course #	Sec.	Course Title	Place
8:00-10:00	4634	A	FINANCIAL INSTITUT.	LRC21

Time	Course #	Sec.	Course Title	Place
10:30-12:30	1221	A	CHEM.PRINCIPLES I	LRC21
	1729-1	A	SELF INSTR- BASIC	N.GR.F.4
	1744	A	TRANSLATORS &LOADERS	Sg.12
	2232	A	EXCEPTIONAL CHILD	Sg.6
	2326	A	HIST.POLIT.THOUGHT II	Sg.27
	4121	A	MICRO ECONOMICS	N.113
	4121	B	MICRO ECONOMICS	LRC21
	4121	C	MICRO ECONOMICS	O.G.B.
	4121	D	MICRO ECONOMICS	N.206
	4121	E	MICRO ECONOMICS	N.208
	4121	F	MICRO ECONOMICS	N.GR.F.2
	4431	A	INTL.MARKETING	N.GR.F.3
	4431	B	INTL.MARKETING	Sg.10
	4432	A	MARKETING PROBLEMS	Sg.12
	5143	A	MODERN ART	O.G.B.
	5337	A	ART OF THE FILM	N.GR.FL
	5337	B	ART OF THE FILM	Sg.29

Time	Course #	Sec.	Course Title	Place
1:00-3:00	1123	A	MICROBIOLOGY	N.206
	1333	A	MULTIDIMEN.CALCULUS	LRC21
	4823	A	MGT.INFO.SYSTEMS	LRC21
	4823	B	MGT.INFO.SYSTEMS	Sg.12
	4823	C	MGT.INFO.SYSTEMS	O.G.B.
	5328	A	STUD.COMMUNICATION	O.G.B.
	5427	A	CHORALE	Sg.10

Time	Course #	Sec.	Course Title	Place
3:30-5:30	1324	A	LINEAR ALGEBRA	LRC21
	2342	A	M.E.IN INTL.AFFAIRS	Sg.12
	3132	A	GUIDANCE &COUNSELING	N.GR.FL
	3337	A	TEACH.READING-ELEM.	N.GR.F.1
	3426	A	STANDARD ARABIC II	O.G.B.
	3426	B	STANDARD ARABIC II	N.213
	4241	A	AUDITING	LRC21
	4640	A	SECURITY ANALYSIS	N.208
	5532	A	POETRY	N.113
	5611	A	ARAB.ESSAY RD& WR.I	Sg.29
	5611	C	ARAB.ESSAY RD& WR.I	O.G.B.
	5611	D	ARAB.ESSAY RD& WR.I	N.GR.FL
	5611	E	ARAB.ESSAY RD& WR.I	Sg.12
	5612	A	ARAB.ESSAY RD& WR.II	N.GR.F.4
	5612	B	ARAB.ESSAY RD& WR.II	Sg.10
	5612	C	ARAB.ESSAY RD& WR.II	N.GR.F.2

Final Exams Schedule (FESP Work)
SPRING 94-95

Tuesday 13, Jun 1995

Time	Course #	Sec.	Course Title	Place
10:30-12:30	1312	A	CALCULUS II	Sg.27
	1312	B	CALCULUS II	Sg.10
	1321	A	CALCULUS III	N.GR.FL
	1321	B	CALCULUS III	O.G.B.
	3137	A	EDUCATION.TECHNOLOGY	N.206
	4122	A	MACRO ECONOMICS	N.208
	4122	B	MACRO ECONOMICS	Sg.12
	4122	C	MACRO ECONOMICS	LRC21
	4122	D	MACRO ECONOMICS	LRC21
	4132	A	MON.THEORY & POLICY	O.G.B.
	5924	A	PHOTOGRAPHY I	N.113
	5924	B	PHOTOGRAPHY I	Sg.12
	5924	C	PHOTOGRAPHY I	Sg.6

Time	Course #	Sec.	Course Title	Place
1:00-3:00	1135	A	GENETICS	Sg.29
	1322	A	BUSINESS STATISTICS	Sg.12
	1322	B	BUSINESS STATISTICS	O.G.B.
	1322	C	BUSINESS STATISTICS	LRC21
	1322	D	BUSINESS STATISTICS	Sg.6
	1327	A	APPLIED STATISTICS	LRC21
	1327	B	APPLIED STATISTICS	Sg.10
	1738	A	COMP.ORGANIZATION	N.206
	1742	A	OPERATIONS RESEARCH	O.G.B.
	4342	A	HUMAN RESOURCE DEV.	Sg.27
	4531	A	INTERM.BUS.STATISTIC	N.208
	4531	B	INTERM.BUS.STATISTIC	N.113
	4531	C	INTERM.BUS.STATISTIC	N.GR.FL
	5334	A	CREATIVE DRAMATICS	Sg.12

Time	Course #	Sec.	Course Title	Place
3:30-5:30	1134	A	ECOLOGY	N.GR.FL
	1247	A	INSTRUMENT.CHEM.ANAL	Sg.29
	1611	A	BASIC HEALTH	Sg.12
				F.A.2
				F.A.2
				F.A.2
				F.A.2
				F.A.2
				F.A.2
				F.A.2
				F.A.2
	1611	B	BASIC HEALTH	LRC21
				F.A.2
				F.A.2
				F.A.2
				F.A.2
				F.A.2
	1729-2	A	SELF INSTR- FORTRAN	N.GR.F.2

Final Exams Schedule (FESP Work)
SPRING 94-95

Tuesday 13, Jun 1995

Course #	Sec.	Course Title	Place
5731	A	EUR.HIST.SINCE 1914	N.113
5831	A	WOMEN IN ARAB WORLD	O.G.B.
7388-1	A	TOP:INTL ORGANIZATION	Sg.6
8791	A	SOFTWARE ENGINEERING	N.208
9381	A	MANAGEMENT THEORY	Sg.10
9387	A	INTERNATIONAL BUS.	O.G.B.
ELE 301	A	ELECTRICAL CIRCUITS I	Sg.27
PHA 303	A	PHARMAC.CALCULATIONS	N.206

Final Exams Schedule (FESP Work)
SPRING 94-95

Thursday 15, Jun 1995

Time	Course #	Sec.	Course Title	Place
8:00-10:00	1328	A	BASIC MATHEMATICS	LRC21
	1328	B	BASIC MATHEMATICS	Sg.12
	1328	C	BASIC MATHEMATICS	O.G.B.
	1732	A	FILE STRUCT.& PROC.	O.G.B.
	1732	B	FILE STRUCT.& PROC.	N.206
	4331	A	ORG.BEHAVIOR	LRC21
	4331	B	ORG.BEHAVIOR	Sg.10

Time	Course #	Sec.	Course Title	Place
10:30-12:30	1211	A	GENERAL CHEMISTRY	N.208
	1325	A	DISCRETE MATHEMATICS	Sg.12
	1325	C	DISCRETE MATHEMATICS	Sg.12
	2337	A	PUBLIC INTL.LAW	Sg.10
	4421	A	INTROD.TO MARKETING	O.G.B.
	4421	B	INTROD.TO MARKETING	N.206
	4421	C	INTROD.TO MARKETING	LRC21
	4441	A	MARKETING RESEARCH	LRC21
	5318	A	INTROD.TO RD/TV/FILM	Sg.6
	5318	B	INTROD.TO RD/TV/FILM	N.113
	5318	C	INTROD.TO RD/TV/FILM	N.GR.FL

Time	Course #	Sec.	Course Title	Place
1:00-3:00	1122	A	GENERAL BOTANY	N.GR.FL
	1223	A	CHEMICAL ANALYSIS I	Sg.6
	1422	A	ELECTRICITY & MAGNET	LRC21
	1736	A	DATA STRUCT.& ALGOR.	LRC21
	2434	A	PERCEPTION	Sg.27
	3331	A	LANGUAGE ARTS	Sg.12
	4521	A	BUSINESS MATHEMATICS	Sg.12
	4521	B	BUSINESS MATHEMATICS	Sg.12
	5111	A	INTROD.MUSIC & ART	O.G.B.
	5111	B	INTROD.MUSIC & ART	O.G.B.
	5111	C	INTROD.MUSIC & ART	Sg.10
	5111	D	INTROD.MUSIC & ART	LRC21
	5111	E	INTROD.MUSIC & ART	N.206
	5111	F	INTROD.MUSIC & ART	N.208
	5111	G	INTROD.MUSIC & ART	N.113
	5925	A	PHOTOGRAPHY II	O.G.B.

Time	Course #	Sec.	Course Title	Place
3:30-5:30	1121	A	GENERAL ZOOLOGY	N.206
	1133	A	VERT.ANAT.& PHYS.	N.GR.FL
	2322	A	INTROD.PUB.ADMIN.	LRC21
	4633	A	MANAGERIAL FINANCE	LRC21
	4633	B	MANAGERIAL FINANCE	N.113
	4633	C	MANAGERIAL FINANCE	Sg.12
	4633	D	MANAGERIAL FINANCE	Sg.12

Final Exams Schedule (FESP Work)
SPRING 94-95

Thursday 15, Jun 1995

Course #	Sec.	Course Title	Place
4821	A	INTROD.TO BUSINESS	Sg.10
4821	B	INTROD.TO BUSINESS	O.G.B.
4821	C	INTROD.TO BUSINESS	O.G.B.
5323	B	INTROD.TECH.STAGE CR.	Sg.27
5324	A	MASS COMM.ESSENTIALS	N.208
5341-2	B	PLAY PRODUCTION I	Sg.6

Final Exams Schedule (FESP Work)
SPRING 94-95

Friday 16, Jun 1995

Time	Course #	Sec.	Course Title	Place
8:00-10:00	2521	A	INTROD.TO SOCIOLOGY	LRC21
	2521	B	INTROD.TO SOCIOLOGY	Sg.12
	4242	A	ADVANCED ACCOUNTING	O.G.B.

Time	Course #	Sec.	Course Title	Place
10:30-12:30	1011	A	INTROD.BIOL.SCIENCE	LRC21
	1011	B	INTROD.BIOL.SCIENCE	N.208
	1012	A	INTROD.PHYS.SCIENCE	LRC21
	1012	B	INTROD.PHYS.SCIENCE	N.206
	1421	A	MECHANICS	N.113
	4321	A	INTROD.TO MANAGEMENT	O.G.B.
	4321	B	INTROD.TO MANAGEMENT	Sg.10
	4321	C	INTROD.TO MANAGEMENT	Sg.12
	5637	A	MEDIA IN MIDDLE EAST	Sg.6

Time	Course #	Sec.	Course Title	Place
1:00-3:00	1311	A	CALCULUS I	N.GR.FL
	1432	A	INTROD.MODERN PHYS.	Sg.12
	1745	A	OPER.SYST.&COMP.ARCH.	O.G.B.
	2421	A	INTROD.TO PSYCHOLOGY	O.G.B.
	2421	B	INTROD.TO PSYCHOLOGY	LRC21
	2421	C	INTROD.TO PSYCHOLOGY	LRC21
	2421	D	INTROD.TO PSYCHOLOGY	N.206
	2421	E	INTROD.TO PSYCHOLOGY	Sg.12
	2421	F	INTROD.TO PSYCHOLOGY	N.208
	2446	A	PSYCH.OF LEARNING	Sg.10
	2534	A	SOCIOL.OF ARAB WORLD	N.GR.F.3
	4131	A	MANAGERIAL ECONOMICS	N.113
	4131	B	MANAGERIAL ECONOMICS	Sg.27
	5432	A	SURVEY WESTERN MUSIC	N.GR.F.2
	7380	A	THEOR.INTL.RELATIONS	N.GR.F.1
	8784	A	COMPUTER NETWORKS	N.GR.F.4
	8794	A	ARTIF.INTELLIGENCE	N.213
	9181	A	BUSINESS ECONOMICS	Sg.6
	9881-2	A	SEM:COM.BKG.&MGT.POL	Sg.29
	CIE 301	A	STRENGTH OF MATES	N.215

Time	Course #	Sec.	Course Title	Place
3:30-5:30	1720	A	COMPUTER LITERACY	N.113
	1720	B	COMPUTER LITERACY	N.206
	1720	C	COMPUTER LITERACY	Sg.10
	1720	D	COMPUTER LITERACY	Sg.6
	1720	E	COMPUTER LITERACY	LRC21
	1722	A	INTROD.TO COMP.PROG.	Sg.12
	1722	B	INTROD.TO COMP.PROG.	N.208
	1725	A	ADVANCED COMP.PROG.	O.G.B.
	1725	B	ADVANCED COMP.PROG.	N.GR.FL

Final Exams Schedule (FESP Work)
SPRING 94-95

Friday 16, Jun 1995

Course #	Sec.	Course Title	Place
1734	A	SYST.ANAL.& DESIGN	Sg.27
1734	B	SYST.ANAL.& DESIGN	LRC21
2333	A	GOVT.OF A MAJ.POWER	O.G.B.
2437	A	INFANCY	Sg.12
5632	A	TRANS.& CREAT.WRI.	Sg.29

Final Exams Schedule (FESP Work)
SPRING 94-95

Saturday 17, Jun 1995

Time	Course #	Sec.	Course Title	Place
3:30-5:30	4322	A	PERSONNEL MANAGEMENT	LRC21
	4322	B	PERSONNEL MANAGEMENT	Sg.12

Final Exams Schedule (FESP Work)
SPRING 94-95

Monday 19, Jun 1995

Time	Course #	Sec.	Course Title	Place
8:00-10:00	4822	A	BUS.COMMUNICATION	LRC21
	4822	B	BUS.COMMUNICATION	Sg.12
	4822	C	BUS.COMMUNICATION	O.G.B.

Time	Course #	Sec.	Course Title	Place
10:30-12:30	3122	A	FUND.OF EDUCATION	Sg.12
	3122	B	FUND.OF EDUCATION	LRC21
	5411	A	INTROD.TO PHILOSOPHY	LRC21
	5411	B	INTROD.TO PHILOSOPHY	O.G.B.
	5411	C	INTROD.TO PHILOSOPHY	Sg.10

Time	Course #	Sec.	Course Title	Place
1:00-3:00	1149	A	BIOLOGY SEN.STUDY	N.215
	1431	A	DYNAMICS	N.GR.F.2
	1724	A	LINEAR PROGRAMMING	Sg.29
	1746	A	DATA BASE SYSTEMS	O.G.B.
	2047	A	RESEARCH DESIGN METH	N.213
	2321	A	INTROD.POLIT.SCIENCE	O.G.B.
	2321	B	INTROD.POLIT.SCIENCE	Sg.12
	2321	C	INTROD.POLIT.SCIENCE	N.113
	2321	D	INTROD.POLIT.SCIENCE	LRC21
	2321	E	INTROD.POLIT.SCIENCE	N.206
	2321	F	INTROD.POLIT.SCIENCE	N.208
	2334	A	CONCEPTS INTL.REL.	LRC21
	2438	A	COGNITIVE DEVELOP.	N.GR.F.4
	3138	A	TESTING MEAS.& EVAL.	Sg.12
	4343	A	PROJECT MANAGEMENT	N.GR.FL
	5341-1	A	PLAY PRODUCTION I	Sg.27
	9583	A	QUANT.METHODS BUS.	Sg.6
	9586	A	RESEARCH METHODS	N.GR.F.3
	PHA 302	A	HIST.& ETHIC.OF PHARM	Sg.10

Time	Course #	Sec.	Course Title	Place
3:30-5:30	1222	A	CHEM.PRINCIPLES II	N.215
	2241	A	LAB.EXPERIENCE ECE	N.GR.F.1
	2642	A	SOCIAL WORK METHOD.	Sg.10
	5421	A	CULTURAL STUDIES I	Sg.12
	5421	B	CULTURAL STUDIES I	O.G.B.
	5421	C	CULTURAL STUDIES I	LRC21
	5421	D	CULTURAL STUDIES I	O.G.B.
	5421	E	CULTURAL STUDIES I	LRC21
	5421	F	CULTURAL STUDIES I	Sg.10
	5421	G	CULTURAL STUDIES I	N.213
	5422	A	CULTURAL STUDIES II	Sg.29
	5422	B	CULTURAL STUDIES II	N.206
	5422	C	CULTURAL STUDIES II	Sg.6
	5422	D	CULTURAL STUDIES II	Sg.12

Final Exams Schedule (FESP Work)
SPRING 94-95

Monday 19, Jun 1995

Course #	Sec.	Course Title	Place
5422	E	CULTURAL STUDIES II	N.GR.FL
5422	F	CULTURAL STUDIES II	N.208
5422	G	CULTURAL STUDIES II	Sg.27
5431	A	CULTURAL STUDIES III	N.GR.F.4
5431	B	CULTURAL STUDIES III	N.113
5431	C	CULTURAL STUDIES III	N.GR.F.3
5431	E	CULTURAL STUDIES III	N.GR.F.1
5431	F	CULTURAL STUDIES III	N.215
5528	A	SURVEY ENG.LIT. II	LRC21
9884	A	BUS.POLICY &PLANNING	N.GR.F.2

Appendix B

Final Exams Schedule (Manual Work)
SPRING 94-95

Saturday 10, Jun 1995

Time	Course #	Sec.	Course Title	Place
8:00-10:00	2820	A	LEARN.RESOURCES TECH	LRC21
	2820	B	LEARN.RESOURCES TECH	LRC21
	2820	C	LEARN.RESOURCES TECH	LRC21
	2820	D	LEARN.RESOURCES TECH	O.G.B.
	2820	E	LEARN.RESOURCES TECH	O.G.B.
	2820	F	LEARN.RESOURCES TECH	LRC21
	2820	G	LEARN.RESOURCES TECH	O.G.B.
	2820	H	LEARN.RESOURCES TECH	LRC21
	2820	I	LEARN.RESOURCES TECH	Sg.10
	2820	J	LEARN.RESOURCES TECH	Sg.10
	2820	K	LEARN.RESOURCES TECH	Sg.12
	2820	L	LEARN.RESOURCES TECH	Sg.12
	2820	M	LEARN.RESOURCES TECH	Sg.12

Time	Course #	Sec.	Course Title	Place
10:30-12:30	5427	A	CHORALE	Sg.10

Time	Course #	Sec.	Course Title	Place
1:00-3:00	1611	A	BASIC HEALTH	Sg.10
				Sg.12
				Sg.6
	1611	B	BASIC HEALTH	N.GR.F.2
				N.GR.F.3
				N.GR.F.4
				N.GR.FL
				O.G.B.

Final Exams Schedule (Manual Work)
SPRING 94-95

Monday 12, Jun 1995

Time	Course #	Sec.	Course Title	Place
8:00-10:00	5510	A	ENGLISH I	Sg.10
	5510	B	ENGLISH I	Sg.10
	5510	C	ENGLISH I	Sg.27
	5510	D	ENGLISH I	O.G.B.
	5510	E	ENGLISH I	O.G.B.
	5510	F	ENGLISH I	Sg.12
	5510	G	ENGLISH I	N.GR.F.1
	5510	H	ENGLISH I	Sg.12
	5511-2	A	ENGLISH II	Sg.29
	5511-2	B	ENGLISH II	N.215
	5511-2	C	ENGLISH II	N.206
	5511-2	D	ENGLISH II	N.GR.F.1
	5511-2	E	ENGLISH II	O.G.B.
	5511-2	F	ENGLISH II	LRC21
	5511-2	G	ENGLISH II	O.G.B.
	5511-2	H	ENGLISH II	LRC21
	5511-2	I	ENGLISH II	Sg.8
	5521	A	COMMUNICATION ART	Sg.12
	5521	B	COMMUNICATION ART	N.208
	5521	C	COMMUNICATION ART	N.GR.F.2
	5521	D	COMMUNICATION ART	N.GR.F.4
	5521	E	COMMUNICATION ART	LRC21
	5521	F	COMMUNICATION ART	N.215
	5521	G	COMMUNICATION ART	LRC21
	5632	A	TRANS.& CREAT.WRI.	Sg.7

Time	Course #	Sec.	Course Title	Place
10:30-12:30	1135	A	GENETICS	Sg.29
	1322	A	BUSINESS STATISTICS	O.G.B.
	1322	B	BUSINESS STATISTICS	O.G.B.
	1322	C	BUSINESS STATISTICS	N.GR.FL
	1322	D	BUSINESS STATISTICS	LRC21
	1327	A	APPLIED STATISTICS	Sg.12
	1327	B	APPLIED STATISTICS	Sg.12
	1336	A	STATISTICS	Sg.10
	1742	A	OPERATIONS RESEARCH	Sg.27
	2241	A	LAB.EXPERIENCE ECE	Sg.5
	2333	A	GOVT.OF A MAJ.POWER	Sg.6
	2434	A	PERCEPTION	Sg.8
	4531	A	INTERM.BUS.STATISTIC	LRC21
	4531	B	INTERM.BUS.STATISTIC	N.215
	4531	C	INTERM.BUS.STATISTIC	LRC21
	ELE 301	A	ELECTRICAL CIRCUITS I	Sg.7

Time	Course #	Sec.	Course Title	Place
1:00-3:00	1123	A	MICROBIOLOGY	Sg.27
	5512-1	A	ENGLISH III	LRC21
	5512-1	B	ENGLISH III	LRC21
	5512-1	C	ENGLISH III	O.G.B.

Final Exams Schedule (Manual Work)
SPRING 94-95

Monday 12, Jun 1995

Course #	Sec.	Course Title	Place
5512-1	D	ENGLISH III	Sg.10
5512-1	E	ENGLISH III	N.206 O.G.B.
5512-1	F	ENGLISH III	N.113
5512-1	G	ENGLISH III	Sg.12
5512-1	H	ENGLISH III	Sg.12
5512-1	I	ENGLISH III	Sg.29
5522	A	SOPHOMORE RHETORIC	Sg.10
5522	B	SOPHOMORE RHETORIC	O.G.B.
5522	C	SOPHOMORE RHETORIC	LRC21
5522	D	SOPHOMORE RHETORIC	N.208
5522	E	SOPHOMORE RHETORIC	N.213
5522	F	SOPHOMORE RHETORIC	O.G.B.
5522	G	SOPHOMORE RHETORIC	Sg.12
5522	H	SOPHOMORE RHETORIC	N.215
5522	I	SOPHOMORE RHETORIC	Sg.6
5522	J	SOPHOMORE RHETORIC	LRC21
5532	A	POETRY	Sg.8

Time	Course #	Sec.	Course Title	Place
3:30-5:30	5111	A	INTROD.MUSIC & ART	LRC21
	5111	B	INTROD.MUSIC & ART	O.G.B.
	5111	C	INTROD.MUSIC & ART	LRC21
	5111	D	INTROD.MUSIC & ART	O.G.B.
	5111	E	INTROD.MUSIC & ART	LRC21
	5111	F	INTROD.MUSIC & ART	O.G.B.
	5111	G	INTROD.MUSIC & ART	LRC21
	5731	A	EUR.HIST.SINCE 1914	Sg.8
	5831	A	WOMEN IN ARAB WORLD	Sg.7
	9181	A	BUSINESS ECONOMICS	Sg.12
	9881-2	A	SEM:COM.BKG.&MGT.POL	Sg.12

Final Exams Schedule (Manual Work)
SPRING 94-95

Tuesday 13, Jun 1995

Time	Course #	Sec.	Course Title	Place
8:00-10:00	1121	A	GENERAL ZOOLOGY	Sg.10
	2326	A	HIST.POLIT.THOUGHT II	Sg.6
	2642	A	SOCIAL WORK METHOD.	Sg.5
	3337	A	TEACH.READING-ELEM.	N.113
	4121	A	MICRO ECONOMICS	N.206
	4121	B	MICRO ECONOMICS	N.215
	4121	C	MICRO ECONOMICS	LRC21
	4121	D	MICRO ECONOMICS	N.213
	4121	E	MICRO ECONOMICS	LRC21
	4121	F	MICRO ECONOMICS	N.208
	4421	A	INTROD.TO MARKETING	N.GR.FL
	4421	B	INTROD.TO MARKETING	N.GR.F.2
	4421	C	INTROD.TO MARKETING	N.GR.F.4
	4431	A	INTL.MARKETING	O.G.B.
	4431	B	INTL.MARKETING	O.G.B.
	5321	A	INTROD.DRAMATIC ART	Sg.12
	5321	B	INTROD.DRAMATIC ART	Sg.12

Time	Course #	Sec.	Course Title	Place
10:30-12:30	1122	A	GENERAL BOTANY	Sg.27
	5422	A	CULTURAL STUDIES II	N.206
	5422	B	CULTURAL STUDIES II	LRC21
	5422	C	CULTURAL STUDIES II	N.215
	5422	D	CULTURAL STUDIES II	Sg.12
	5422	E	CULTURAL STUDIES II	Sg.12
	5422	F	CULTURAL STUDIES II	LRC21
	5422	G	CULTURAL STUDIES II	N.213
	5431	A	CULTURAL STUDIES III	O.G.B.
	5431	B	CULTURAL STUDIES III	Sg.10
	5431	C	CULTURAL STUDIES III	O.G.B.
	5431	E	CULTURAL STUDIES III	Sg.10
	5431	F	CULTURAL STUDIES III	O.G.B.

Time	Course #	Sec.	Course Title	Place
1:00-3:00	3331	A	LANGUAGE ARTS	N.206
	5411	A	INTROD.TO PHILOSOPHY	LRC21
	5411	B	INTROD.TO PHILOSOPHY	Sg.10
	5411	C	INTROD.TO PHILOSOPHY	LRC21
	5421	A	CULTURAL STUDIES I	Sg.12
	5421	B	CULTURAL STUDIES I	N.215
	5421	C	CULTURAL STUDIES I	Sg.12
	5421	D	CULTURAL STUDIES I	N.213
	5421	E	CULTURAL STUDIES I	O.G.B.
	5421	F	CULTURAL STUDIES I	O.G.B.
	5421	G	CULTURAL STUDIES I	N.208

Final Exams Schedule (Manual Work)
SPRING 94-95

Tuesday 13, Jun 1995

Time	Course #	Sec.	Course Title	Place
3:30-5:30	2321	A	INTROD.POLIT.SCIENCE	O.G.B.
	2321	B	INTROD.POLIT.SCIENCE	O.G.B.
	2321	C	INTROD.POLIT.SCIENCE	LRC21
	2321	D	INTROD.POLIT.SCIENCE	LRC21
	2321	E	INTROD.POLIT.SCIENCE	N.GR.FL
	2321	F	INTROD.POLIT.SCIENCE	N.GR.F.4
	2334	A	CONCEPTS INTL.REL.	LRC21
	7380	A	THEOR.INTL.RELATIONS	Sg.6
	8791	A	SOFTWARE ENGINEERING	Sg.10
	9283	A	MGT ACCOUNTING	Sg.12
	9881-6	A	SEM:INTL.& GLOB.MKT.	Sg.12

Final Exams Schedule (Manual Work)
SPRING 94-95

Thursday 15, Jun 1995

Time	Course #	Sec.	Course Title	Place
8:00-10:00	1247	A	INSTRUMENT.CHEM.ANAL	Sg.8
	1326	A	DIFFERENT.EQUATIONS	Sg.27
	1735	A	INTROD.COMP.SYSTEMS	Sg.29
	4223	A	PRIN.ACCOUNTING I	Sg.12
	4223	B	PRIN.ACCOUNTING I	N.215
	4223	C	PRIN.ACCOUNTING I	O.G.B.
	4223	D	PRIN.ACCOUNTING I	O.G.B.
	4223	E	PRIN.ACCOUNTING I	Sg.12
	4223	F	PRIN.ACCOUNTING I	N.213
	4321	A	INTROD.TO MANAGEMENT	LRC21
	4321	B	INTROD.TO MANAGEMENT	LRC21
	4321	C	INTROD.TO MANAGEMENT	Sg.10
	4331	A	ORG.BEHAVIOR	N.206
	4331	B	ORG.BEHAVIOR	N.208
	4342	A	HUMAN RESOURCE DEV.	Sg.6

Time	Course #	Sec.	Course Title	Place
10:30-12:30	3426	A	STANDARD ARABIC II	Sg.10
	3426	B	STANDARD ARABIC II	Sg.10
	5528	A	SURVEY ENG.LIT. II	Sg.8
	5611	A	ARAB.ESSAY RD& WR.I	LRC21
	5611	C	ARAB.ESSAY RD& WR.I	Sg.27
	5611	D	ARAB.ESSAY RD& WR.I	Sg.6
	5611	E	ARAB.ESSAY RD& WR.I	Sg.29
	5612	A	ARAB.ESSAY RD& WR.II	Sg.12
	5612	B	ARAB.ESSAY RD& WR.II	O.G.B.
	5612	C	ARAB.ESSAY RD& WR.II	O.G.B.
	5612	D	ARAB.ESSAY RD& WR.II	N.206
	5623	A	APPR.OF ARABIC LIT.	LRC21
	5623	B	APPR.OF ARABIC LIT.	LRC21
	5623	C	APPR.OF ARABIC LIT.	N.GR.FL
	5623	D	APPR.OF ARABIC LIT.	N.208
	5623	E	APPR.OF ARABIC LIT.	Sg.12
	5623	F	APPR.OF ARABIC LIT.	N.215
	5623	G	APPR.OF ARABIC LIT.	N.213
	5623	H	APPR.OF ARABIC LIT.	Sg.12
	5623	I	APPR.OF ARABIC LIT.	N.GR.FL
	5623	J	APPR.OF ARABIC LIT.	N.113
	5637	A	MEDIA IN MIDDLE EAST	Sg.7

Time	Course #	Sec.	Course Title	Place
1:00-3:00	1729-1	A	SELF INSTR- BASIC	Sg.10
	1744	A	TRANSLATORS &LOADERS	Sg.10
	2232	A	EXCEPTIONAL CHILD	N.215
	4122	A	MACRO ECONOMICS	O.G.B.
	4122	B	MACRO ECONOMICS	LRC21
	4122	C	MACRO ECONOMICS	Sg.12
	4122	D	MACRO ECONOMICS	LRC21
	4131	A	MANAGERIAL ECONOMICS	O.G.B.

Final Exams Schedule (Manual Work)
SPRING 94-95

Thursday 15, Jun 1995

Course #	Sec.	Course Title	Place
4131	B	MANAGERIAL ECONOMICS	O.G.B.
5324	A	MASS COMM.ESSENTIALS	N.GR.FL
5341-1	A	PLAY PRODUCTION I	Sg.29
5341-2	B	PLAY PRODUCTION I	Sg.27

Time	Course #	Sec.	Course Title	Place
3:30-5:30	1012	A	INTROD.PHYS.SCIENCE	O.G.B.
	1012	B	INTROD.PHYS.SCIENCE	O.G.B.
	2521	A	INTROD.TO SOCIOLOGY	LRC21
	2521	B	INTROD.TO SOCIOLOGY	LRC21
	2534	A	SOCIOL.OF ARAB WORLD	LRC21
	5842	A	REP.WOMEN ART.& MED.	Sg.8
	7388-1	A	TOP:INTL ORGANIZATION	Sg.6
	8782	A	DATABASE MGT SYST	Sg.10
	9583	A	QUANT.METHODS BUS.	Sg.12
	9586	A	RESEARCH METHODS	Sg.12

Final Exams Schedule (Manual Work)
SPRING 94-95

Friday 16, Jun 1995

Time	Course #	Sec.	Course Title	Place
8:00-10:00	1133	A	VERT.ANAT.& PHYS.	Sg.27
	1223	A	CHEMICAL ANALYSIS I	Sg.29
	1432	A	INTROD.MODERN PHYS.	Sg.8
	3132	A	GUIDANCE &COUNSELING	Sg.7
	4224	A	PRIN.ACCOUNTING II	N.215
	4224	B	PRIN.ACCOUNTING II	Sg.12
	4224	C	PRIN.ACCOUNTING II	Sg.12
	4224	D	PRIN.ACCOUNTING II	LRC21
	4224	E	PRIN.ACCOUNTING II	LRC21
	4633	A	MANAGERIAL FINANCE	O.G.B.
	4633	B	MANAGERIAL FINANCE	N.213
	4633	C	MANAGERIAL FINANCE	LRC21
	4633	D	MANAGERIAL FINANCE	O.G.B.
	4634	A	FINANCIAL INSTITUT.	Sg.10
	4636	A	FINANCIAL MARKETS	Sg.6

Time	Course #	Sec.	Course Title	Place
10:30-12:30	1720	A	COMPUTER LITERACY	LRC21
	1720	B	COMPUTER LITERACY	LRC21
	1720	C	COMPUTER LITERACY	Sg.27
	1720	D	COMPUTER LITERACY	Sg.29
	1720	E	COMPUTER LITERACY	LRC21
	1722	A	INTROD.TO COMP.PROG.	Sg.12
	1722	B	INTROD.TO COMP.PROG.	Sg.12
	1725	A	ADVANCED COMP.PROG.	N.206
	1725	B	ADVANCED COMP.PROG.	Sg.6 Sg.10
	1729-2	A	SELF INSTR- FORTRAN	Sg.12
	1729-6	A	SELF INSTR- C LANG	Sg.12
	1732	A	FILE STRUCT.& PROC.	O.G.B.
	1732	B	FILE STRUCT.& PROC.	O.G.B.
	1734	A	SYST.ANAL.& DESIGN	Sg.8
	1734	B	SYST.ANAL.& DESIGN	O.G.B.
	5143	A	MODERN ART	F.A.2

Time	Course #	Sec.	Course Title	Place
1:00-3:00	1011	A	INTROD.BIOL.SCIENCE	Sg.12
	1011	B	INTROD.BIOL.SCIENCE	Sg.12
	1211	A	GENERAL CHEMISTRY	O.G.B.
	1232	A	ORGANIC CHEMISTRY II	O.G.B.
	1421	A	MECHANICS	Sg.27
	1724	A	LINEAR PROGRAMMING	Sg.29
	1745	A	OPER.SYST.&COMP.ARCH.	Sg.10
	2337	A	PUBLIC INTL.LAW	LRC21
	2438	A	COGNITIVE DEVELOP.	Sg.7
	3147	A	PRAC.TEACH- ELEM.	N.215
	3148	A	PRAC.TEACH- SECOND.	N.215
	4441	A	MARKETING RESEARCH	LRC21
	5432	A	SURVEY WESTERN MUSIC	Sg.6

Final Exams Schedule (Manual Work)
SPRING 94-95

Friday 16, Jun 1995

Course #	Sec.	Course Title	Place
CIE 301	A	STRENGTH OF MATERIALS	Sg.8

Time	Course #	Sec.	Course Title	Place
3:30-5:30	3137	A	EDUCATION.TECHNOLOGY	O.G.B.
	4823	A	MGT.INFO.SYSTEMS	LRC21
	4823	B	MGT.INFO.SYSTEMS	LRC21
	4823	C	MGT.INFO.SYSTEMS	O.G.B.
	5318	A	INTROD.TO RD/TV/FILM	N.206
	5318	B	INTROD.TO RD/TV/FILM	N.208
	5318	C	INTROD.TO RD/TV/FILM	N.215
	7181	A	INTERN.POLIT.ECONOMY	Sg.6
	8784	A	COMPUTER NETWORKS	Sg.10
	9884	A	BUS.POLICY &PLANNING	Sg.12

Final Exams Schedule (Manual Work)
SPRING 94-95

Saturday 17, Jun 1995

Time	Course #	Sec.	Course Title	Place
8:00-10:00	1311	A	CALCULUS I	O.G.B.
	1312	A	CALCULUS II	Sg.10
	1312	B	CALCULUS II	Sg.10
	1321	A	CALCULUS III	O.G.B.
	1321	B	CALCULUS III	O.G.B.
	1324	A	LINEAR ALGEBRA	Sg.12
	1333	A	MULTIDIMEN.CALCULUS	Sg.12
	1339	A	NUMERICAL ANALYSIS	Sg.29
	2342	A	M.E.IN INTL.AFFAIRS	N.GR.FL
	3336	A	MODERN ENG.GRAMMAR	Sg.27
	4242	A	ADVANCED ACCOUNTING	N.206
	4521	A	BUSINESS MATHEMATICS	N.208
	4521	B	BUSINESS MATHEMATICS	Sg.5
	5223	A	HIST.ARCH.& FURN. II	LRC21
	5344	A	TV PROD.& DIRECTION	Sg.7
	5344	B	TV PROD.& DIRECTION	Sg.8
	5924	A	PHOTOGRAPHY I	N.215
	5924	B	PHOTOGRAPHY I	N.215
	5924	C	PHOTOGRAPHY I	Sg.6
	5925	A	PHOTOGRAPHY II	N.213

Time	Course #	Sec.	Course Title	Place
10:30-12:30	1221	A	CHEM.PRINCIPLES I	LRC21
	1325	A	DISCRETE MATHEMATICS	Sg.12
	1325	C	DISCRETE MATHEMATICS	Sg.27
	1328	A	BASIC MATHEMATICS	O.G.B.
	1328	B	BASIC MATHEMATICS	O.G.B.
	1328	C	BASIC MATHEMATICS	N.GR.FL
	1736	A	DATA STRUCT.& ALGOR.	Sg.12
	2322	A	INTROD.PUB.ADMIN.	Sg.10
	3138	A	TESTING MEAS.& EVAL.	N.213
	4241	A	AUDITING	N.215
	5337	A	ART OF THE FILM	LRC21
	5337	B	ART OF THE FILM	LRC21

Time	Course #	Sec.	Course Title	Place
1:00-3:00	2421	A	INTROD.TO PSYCHOLOGY	LRC21
	2421	B	INTROD.TO PSYCHOLOGY	O.G.B.
	2421	C	INTROD.TO PSYCHOLOGY	Sg.12
	2421	D	INTROD.TO PSYCHOLOGY	O.G.B.
	2421	E	INTROD.TO PSYCHOLOGY	Sg.12
	2421	F	INTROD.TO PSYCHOLOGY	N.GR.FL
	2446	A	PSYCH.OF LEARNING	Sg.10
	4432	A	MARKETING PROBLEMS	LRC21

Time	Course #	Sec.	Course Title	Place
3:30-5:30	3122	A	FUND.OF EDUCATION	O.G.B.

Final Exams Schedule (Manual Work)
SPRING 94-95

Saturday 17, Jun 1995

Course #	Sec.	Course Title	Place
3122	B	FUND.OF EDUCATION	O.G.B.
4322	A	PERSONNEL MANAGEMENT	Sg.10
4322	B	PERSONNEL MANAGEMENT	LRC21
4635	A	BANKING OPERATIONS	LRC21
8794	A	ARTIF.INTELLIGENCE	Sg.10
9381	A	MANAGEMENT THEORY	Sg.12
9387	A	INTERNATIONAL BUS.	Sg.12

Final Exams Schedule (Manual Work)
SPRING 94-95

Monday 19, Jun 1995

Time	Course #	Sec.	Course Title	Place
8:00-10:00	1134	A	ECOLOGY	Sg.8
	1222	A	CHEM.PRINCIPLES II	Sg.27
	1234	A	QUANTUM CHEMISTRY	Sg.27
	1431	A	DYNAMICS	Sg.29
	2047	A	RESEARCH DESIGN METH	Sg.7
	4236	A	COST ACCOUNTING	O.G.B.
	4236	B	COST ACCOUNTING	O.G.B.
	4640	A	SECURITY ANALYSIS	Sg.5
	4822	A	BUS.COMMUNICATION	Sg.12
	4822	B	BUS.COMMUNICATION	Sg.12
	4822	C	BUS.COMMUNICATION	Sg.12
	5323	B	INTROD.TECH.STAGE CR.	Sg.6
	5334	A	CREATIVE DRAMATICS	Sg.6
	5732	A	HISTORY OF LEBANON	N.GR.FL
	PHA 303	A	PHARMAC.CALCULATIONS	Sg.10

Time	Course #	Sec.	Course Title	Place
10:30-12:30	1149	A	BIOLOGY SEN.STUDY	Sg.29
	1422	A	ELECTRICITY & MAGNET	Sg.10
	1738	A	COMP.ORGANIZATION	Sg.12
	1746	A	DATA BASE SYSTEMS	Sg.12
	2341	A	INTL.REGIONAL ORG.	Sg.8
	2437	A	INFANCY	Sg.6
	4132	A	MON.THEORY & POLICY	N.206
	4343	A	PROJECT MANAGEMENT	N.208
	5328	A	STUD.COMMUNICATION	Sg.27
	PHA 302	A	HIST.& ETHIC.OF PHARM	O.G.B.

Time	Course #	Sec.	Course Title	Place
1:00-3:00	2223	A	PSYCH.OF YOUNG CHILD	O.G.B.
	2223	B	PSYCH.OF YOUNG CHILD	O.G.B.
	4821	A	INTROD.TO BUSINESS	Sg.12
	4821	B	INTROD.TO BUSINESS	Sg.12
	4821	C	INTROD.TO BUSINESS	Sg.10

Time	Course #	Sec.	Course Title	Place
3:30-5:30	4826	A	BUSINESS LAW	O.G.B.
	4826	B	BUSINESS LAW	N.215
	4826	C	BUSINESS LAW	O.G.B.

USER MANUAL

In this chapter represents the **FESP** user manual. It describes the steps to be followed when running **FESP. FESP** main menu and sub-menus are presented in Figure 1.

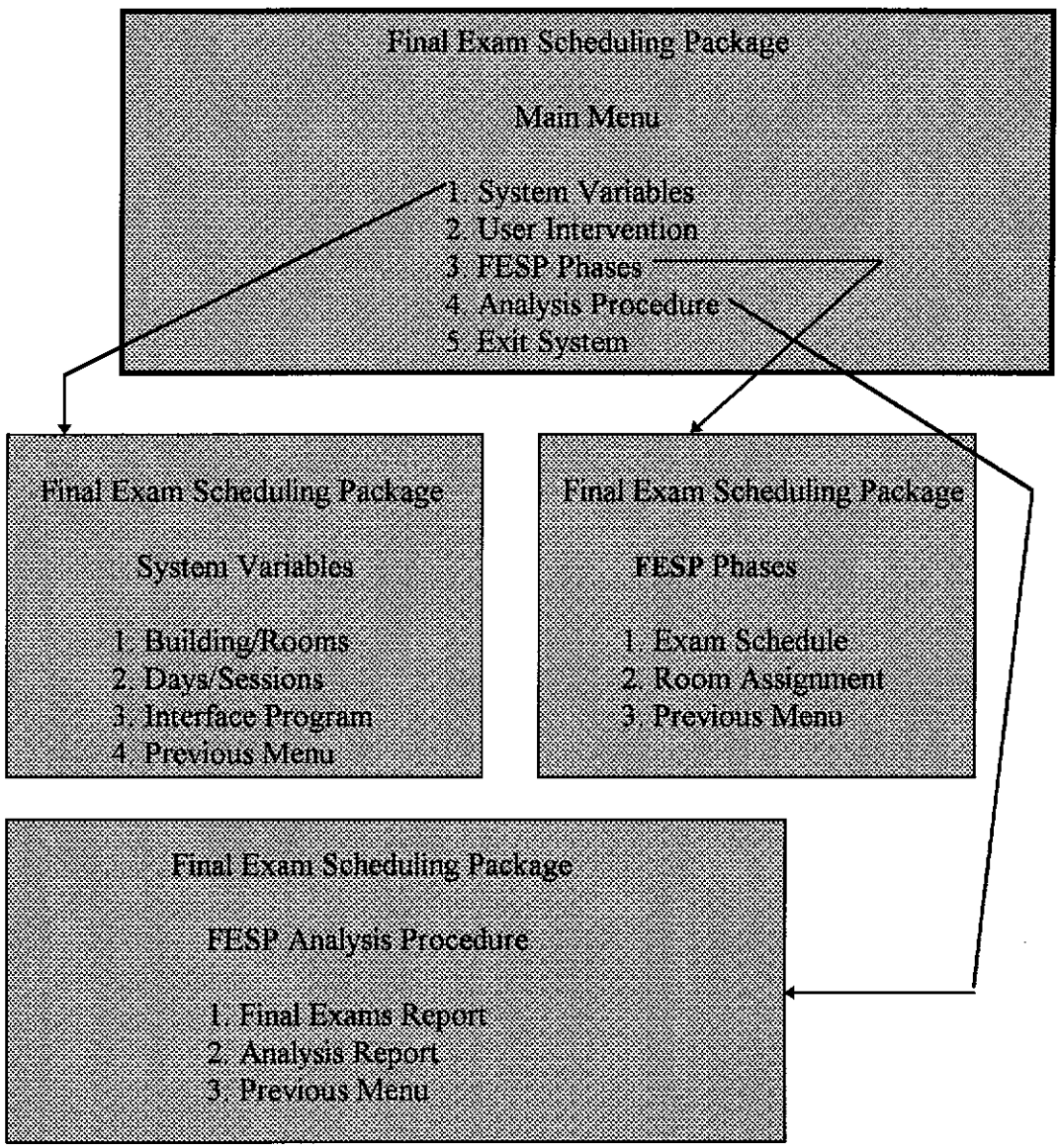


Figure C.1: FESP Main and Sub-Menus.

Let us call the main menu MM, the system variables sub-menu SVM, the phases sub-menu PM, and the analysis sub-menu AM.

- **Option 1 in MM displays SVM where the user can do the following:**

SVM option 1: The user selects a building from a list of values. Then the numbers of the classrooms of that building are displayed. The user flags the classrooms that are to be used in the scheduling process. The user sets a ratio that identifies the classroom available capacity over its seating capacity.

SVM option 2: The user specifies the number of exam days (**ED**), the date of the first exam, and the number of exam sessions per day (**ES**). Then a set of records numbered from 1 to **ED** are displayed. These records represents the exam days. The user flags the reading days. Then a set of records numbered from 1 to **ES** is displayed. These records represents the number of exam sessions per day. The user sets the time of each session.

SVM option 3: The user specifies what data is to be imported into **FESP**. The options are: Course Catalog, Student Schedules, Rooms File and Building File. One, some or all of these files can be specified.

- **Option 2 in MM displays the user intervention screen where the user can do the following:**

The user specifies a group name (*i.e.* a sub-schedule name) and then he/she enters the course numbers, section numbers, session numbers, day numbers, room numbers and the associated building. Such pre-assignments will never be touched by **FESP**. He can choose another group and enters another set of pre-assignments. Finally he selects a one of these groups to be active in the next run of **FESP**.

- **Option 3 in MM displays the PM where the user can do the following:**

PM option 1: By confirming this process, the final exam scheduling will be done.

PM option 2: By confirming this process, the classrooms assignment will be done.

- **Option 4 in MM displays the AM where the user can do the following:**

AM option 1: Produces the final exam schedule lists (report).

AM option 2: Produces the analysis report of the last run of **FESP**.

References

Balakrishnan N., Lucena A., Wong R. (1992) Scheduling Examinations to Reduce Second-Order Conflicts. *Computers Ops. Res. Vol. 19, No. 5, pp. 353-361, 1992.*

Brassard G., and Brately P. 1988. *Algorithmics Theory and Practice*. Prentice-Hall International, Inc.

Lotfi V. and Cerveny R. (1991) A Final-exam-scheduling Package. *J. Opl Res. Soc. Vol. 42, No. 3, pp. 205-216.*

Lucena A. (1990) Time-dependent travelling saleman - the deliveryman case. Networks 20, 753-763.

Mehta N. 1981.. The Application of A Graph Coloring Method to an Examination Scheduling Problem. *Interfaces Vol. 11, No. 5, October 1981.*

Mehta N. 1980. Performance of Selected Graph Coloring Algorithms - Empirical Results, TIMS/ORSA Conference, Washington, May.

Romero, B. 1982. Examination Scheduling in a large Engineering School: A Computer-Assisted Participative Procedure. *Interfaces Vol. 12, No. 2, April 1982.*