ASSUMPTION UNIVERSITY

# COMPUTERIZED INFORMATION SYSTEM FOR PRIVATE SCHOOL

by

MISS LADDAWAN CHONGSIRIRATANAKUL

Final Report of the Three - Credit Course
CS 6998 System Development Project

Submitted In Partial Fulfillment
of the Requirements for the Degree of
Master of Science
in Computer Information Systems
Assumption University

December 1992

110691

ABAC
GRADUATE SCHOOL LIBRARY

# COMPUTERIZED INFORMATION SYSTEM FOR PRIVATE SCHOOL

BY

MISS LADDAWAN CHONGSIRIRATANAKUL

Final Report of the Three-Credit Course
CS 6998 System Development Project

Submitted in Partial Fulfillment
of the Requirements for the Degree of
Master of Science
in Computer Information Systems
Assumption University

December 1992

Project Title : Computerized Information System for Private School

Name : Miss Laddawan Chongsiriratanakul

Project Advisor : Assoc. Prof. Dr. Kanchit Malaivongs

Academic Year : 1992

---

The Graduate School of Assumption Univeristy had approved this final report of the three-credit course, CS 6998 System Development Project, submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Information Systems.

Approval Committee :

(Assoc.Prof.Dr.Kanchit Malaivongs)
Advisor

(Prof.Dr.Srisakdi Charmonman)
Member

(Dr.Sudhiporn Patumtaewapibal)
Member

(Dr.Buranawong Sowaprux)
Member

(Assoc.Prof.Somchai Thayarnyong)
Member

December 1992

ASSUMPTION UNIVERSITY OF THAILAND
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

# ABSTRACT

This system development project is conducted to improve the existing registration, payment and student grading system of the IMAC Institute presently done manually. The key areas to be computerized are creating and maintaining customers, prospect, calculation of cost control in each course, keeping information for business development and report generation. The new system can provide on-line query response, as well as customized reports that are consistent, accurate and relevant.

The methodology used in this project has proved to be successful in producing the system which integrates the business operations with the information service support. The output information is a good strategic planning aid for management in making decision.

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST. GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

# ACKNOWLEDGEMENTS

I wish to acknowledge gratefully the following people for their contributions and assistance in the preparation of this project.

I would like to express my thanks for the many useful comments and suggestion provided by Assoc. Prof. Dr. Kanchit Malaivongs who reviewed for the entired project from start to finish. Without him, this project would certainly never has seen the light of day.

A project of this magnitude could not have been completed smoothly without support. I wish to acknowledge the two persons who help me to finish it on time. Mrs. Pensri Poapongsakorn, who supported me with a computer set from her company during my four-month project. Mr. Somnuek Taichamahamonkol, who gave me his guidance in advance programming techniques.

There are many other people who gave their encourgement but did not want me to mention their names on this page. So, a special debt of gratitude to my family, my friends and my colleagues.

ABAC
GRADUATE SCHOOL LIBRARY

# CONTENTS

Page

TITLE PAGE

ABSTRACT.....i

ACKNOWLEDGEMENTS.....ii

TABLE OF CONTENTS.....iii

LIST OF FIGURES.....v

CHAPTER 1 : INTRODUCTION.....1

1.1 Background.....1

1.2 Objectives of the Project.....2

1.3 Scope of the Project.....3

CHAPTER 2 : EXISTING SYSTEM.....4

2.1 Background of the Organization.....4

2.2 Existing Business Functions.....7

2.3 Current Problems.....9

2.4 Area of Improvement.....10

CHAPTER 3 : PROPOSED SYSTEM.....11

3.1 User Requirements.....11

3.2 Overview of the Proposed System.....11

3.3 Hardware Requirements.....18

# CONTENTS

Page

3.4 Software Requirements.....19

3.5 System Requirements.....20

3.6 Cost and Benefit Analysis.....21

CHAPTER 4 : IMPLEMENTATION PLAN.....24

4.1 Programming.....24

4.2 Testing.....26

4.3 User Training.....27

4.4 Finalizing Documentation.....28

CHAPTER 5 : CONCLUSIONS & RECOMMENDATIONS.....29

5.1 Conclusions.....29

5.2 Recommendations.....31

REFERENCES.....33

APPENDICES : APPENDIX A - Existing Forms.....35

APPENDIX B - Context and Data Flow Diagrams.....43

APPENDIX C - Data Dictionary.....54

APPENDIX D - Data Access Diagram.....63

APPENDIX E - Minispecifications.....65

APPENDIX F - Program Listing (Partial)..74

APPENDIX G - Screen Design.....140

APPENDIX H - Output Reports.....166

# LIST OF FIGURES

Page

Figure 2-1 Organization Chart of "IMAC".....8

Figure 3-1 System Requirements Model for "IMAC".....20

Figure 3-2 Payback Analysis Showing a Payback Period... 23

Figure 4-1 The Hierarchy of IMAC Computerized Information System.....25

Figure A-1 Application Form of the Existing System.....36

Figure A-2 Application Form (Cont'd) of the Existing System.....37

Figure A-3 Registration Form of the Existing System.....38

Figure A-4 Receipt Form of the Existing System.....39

Figure A-5 Grade Report of the Existing System.....40

Figure A-6 Evaluation Form of the Existing System.....41

Figure A-7 Evaluation Form (Cont'd) of the Existing System.....42

Figure B-1 Context Diagram of "IMAC".....44

Figure B-2 DFD Level 0 - A Derived Model of "IMAC".....45

Figure B-3 DFD Level 1 - Process 1 : Build Class File...46

Figure B-4 DFD Level 1 - Process 2 : Build Plan Table File.....47

Figure B-5 DFD Level 1 - Process 3 : Build Master File..48

Figure B-6 DFD Level 1 - Process 4 : Process Registration.....49

Figure B-7 DFD Level 1 - Process 5 : Process Payment....50

Figure B-8 DFD Level 1 - Process 6 : Prepare Class Schedule.....51

# LIST OF FIGURES

Page

Figure B-9 DFD Level 1 - Process 7 : Process Grade.....52

Figure B-10 DFD Level 1 - Process 8 : Monitor Student Progress.....53

Figure C-1 DD of Data Store : Instructor Master File....55

Figure C-2 DD of Data Store : Student Class File.....55

Figure C-3 DD of Data Store : Student Master File.....56

Figure C-4 DD of Data Store : Student Master File (Cont'd).....57

Figure C-5 DD of Data Store : Plan Table File.....58

Figure C-6 DD of Data Store : Student Summary File.....59

Figure C-7 DD of Data Store : Student Transaction File.....60

Figure C-8 DD of Data Store : Control Running File.....61

Figure C-9 Data Dictionary of System Files.....62

Figure D-1 Data Access Diagram Describing Primary Associations between Files.....64

Figure E-1 Minispecification Doc. for Process Bubble 1.0.....66

Figure E-2 Minispecification Doc. for Process Bubble 2.0.....67

Figure E-3 Minispecification Doc. for Process Bubble 3.0.....68

Figure E-4 Minispecification Doc. for Process Bubble 4.0.....69

# LIST OF FIGURES

Page

Figure E-5 Minispecification Doc. for Process Bubble 5.0.....70

Figure E-6 Minispecification Doc. for Process Bubble 6.0.....71

Figure E-7 Minispecification Doc. for Process Bubble 7.0.....72

Figure E-8 Minispecification Doc. for Process Bubble 8.0.....73

Figure G-1 Screen Design of Main Menu.....141

Figure G-2 Screen Design of Menu 'System Setup'.....142

Figure G-3 Screen Design of Plan Table Information.....143

Figure G-4 Screen Design of Instructor Master Information.....144

Figure G-5 Screen Design of Student Class Information.....145

Figure G-6 Screen Design of Student Master Information.....146

Figure G-7 Screen Design of Prospect Master Information.....147

Figure G-8 Screen Design of Menu 'Routine Operation'...148

Figure G-9 Screen Design of Class Cost Control Analysis.....149

Figure G-10 Screen Design of Student Registration (Screen 1).....150

# LIST OF FIGURES

Page

Figure G-11 Screen Design of Student Registration (Screen 2).....151
Figure G-12 Screen Design of Student Registration (Screen 3).....152
Figure G-13 Screen Design of Student Registration (Screen 4).....153
Figure G-14 Screen Design of Student Registration (Screen 5).....154
Figure G-15 Screen Design of Student Registration (Screen 6).....155
Figure G-16 Screen Design of Student Payment (Screen 1).....156
Figure G-17 Screen Design of Student Payment (Screen 2).....157
Figure G-18 Screen Design of Student Grade (Screen 1)...158
Figure G-19 Screen Design of Student Grade (Screen 2)...159
Figure G-20 Screen Design of Student Grade (Screen 3)...160
Figure G-21 Screen Design of Menu 'Month-end and Year-end Works'.....161
Figure G-22 Screen Design of Post Student History (Screen 1).....162
Figure G-23 Screen Design of Post Student History (Screen 2).....163
Figure G-24 Screen Design of Menu 'Reports as Request'..164
Figure G-25 Screen Design of Menu 'Inquiry as Request'..165

# LIST OF FIGURES

Page

Figure H-1 Design of Class Roster Output Form.....167

Figure H-2 Design of Student Transcript Output Form....168

Figure H-3 Design of Course Schedule Output Form.....169

Figure H-4 Design of Class Cost Control Report.....170

Figure H-5 Design of Student Name and Address Run Book.....171

Figure H-6 Design of Student Grade Report.....172

Figure H-7 Design of Student Information by Media Type.....173

Figure H-8 Design of Prospect Information Run Book.....174

Figure H-9 Design of Class-Offering Report.....175

Figure H-10 Design of Tuition Fee for Each Course Schedule.....176

Figure H-11 Design of Student Outlet by Course No.....177

ASSUMPTION UNIVERSITY OF THAILAND
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

# CHAPTER 1
# INTRODUCTION

## 1.1 BACKGROUND

The "IMAC" Institute is a subsidiary of the Technology Supply Group, which is a business partner of IBM (THAILAND) CO., LTD. It has an expertise in developing hospital systems on IBM's mini computers.

"IMAC" was officially established in May 1991 as a vocational computer school under the Ministry of Education.

The main policy of the institute is to train people who do not possess any basics in computer and information technology so that they can work efficiently. This is to help relieve the burdens the government's task in personnel development in information technology.

The second policy is to further develop personnel who are in information technology field so that they can keep up with the technological advancements in this field and use the experiences for self-development.

For the courses to be used in teaching, the institute has chosen "DIPLOMA IN COMPUTER STUDIES" developed by the Institute of National Computing Centre, England. This curriculum is accepted and used as basics in computer personnel development in institutes worldwide, especially in the USA.

Also, the institute has used the course "DIPLOMA IN COMPUTER STUDIES" as basics in formulating the course "DIPLOMA IN BUSINESS COMPUTER".

At present, most administrative work in IMAC is a manual system in which clerks prepare data and reports manually. Some time management must wait for a long time to get reports. Managers also consider to use a computerized system because they want a better communication and strategic planning in the near future. The main target for the new system is to improve reporting and recording-keeping procedure.

## 1.2 OBJECTIVES OF THE PROJECT

This project has the following objectives :

- To analyze, design and develop a computerized system for IMAC in order that the system will be able to provide a convenient access mechanism for keeping historical data, to provide an update information for better future planning.
- To provide effective communication with the clients.
- To support better methodology in reporting to management.

## 1.3 SCOPE OF THE PROJECT

The project will cover all steps of development from analysis of the existing manual system to designing a new computerized system. It will involve :

1. Analyze and design of the registration, payment and student grading system.
2. Develop a new database system that will be useful in reporting to instructors, students and all departments.

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

# CHAPTER 2

# EXISTING SYSTEM

## 2.1 BACKGROUND OF THE ORGANIZATION

The objective of IMAC is to prepare students for practical work in computing and information process. Specifically it aims to :

1. Provide a basic knowledge of computer hardware /software.
2. Introduce the business functions to which computers are applied.
3. Provide introduction to business organization and systems.
4. Provide enough training in programming to make possible the writing and documenting of a program to suit a specific language.
5. Develop the student's skills in communication, oral and written, which play so important a part in business computing and information process.
6. Provide the limited mathematical skills needed for the understanding and use of computers.

IMAC Institute has three major programs of computer courses :

1. Diploma in Computer Studies (NCC)

The syllabus consists of twelve units :

1.1 Computer Concept I

1.2 Computer Programming

1.3 Business Organization

1.4 Use of Application Packages

1.5 Computer Concept II

1.6 COBOL Programming

1.7 Information Management

1.8 Computerized Accounting Techniques

1.9 Computer at Works

1.10 Human Communication

1.11 Computer Related Mathematics and Statistics

1.12 Computer Project

Duration - There are three semesters. Approximately twelve months full-time is needed to complete the whole course.

2. Certificate in Computer Programming (NCC)

The syllabus consists of six units :

2.1 Basic Computer Principles

2.2 RPG Programming

2.3 Use of Application Packages

2.4 Business Background and Communication

2.5 Computer Programming with COBOL

2.6 Computer Project

On the job training courses for the computer departments of various companies are also offered.

Duration - There are two semesters. Approximately six months full-time is needed to complete the whole course.

3. Short Course in Computer Studies (IMAC)

The course is intended to prepare students to work in computing and aims to impart practical business computing skills. The teaching of all units of the course should place due emphasis on :

- Practical assignment
- The computer related aspects
- Use of appropriate application packages
- Integration of the specific units into the computer environment.

Some of the above courses are as follows :

3.1 AS/400 System Operation

3.2 AS/400 Relational Database Design and Coding

3.3 AS/400 Application Programming

3.4 RPG Programming for Beginner

3.5 RPG AS/400 Programming

## 2.2 EXISTING BUSINESS FUNCTIONS

The existing business functions of IMAC are categorized as:

2.2.1 Business Development Division. This is sub-divided into :

(a) Marketing

(b) Administrative

- Accounting and Financce
- Registration and Document
- Purchase and Stock
- General Service

2.2.2 Education Division. This is sub-divided into :

(a) Lab

(b) Full-time Faculty

(c) Part-time Faculty

(d) Library

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

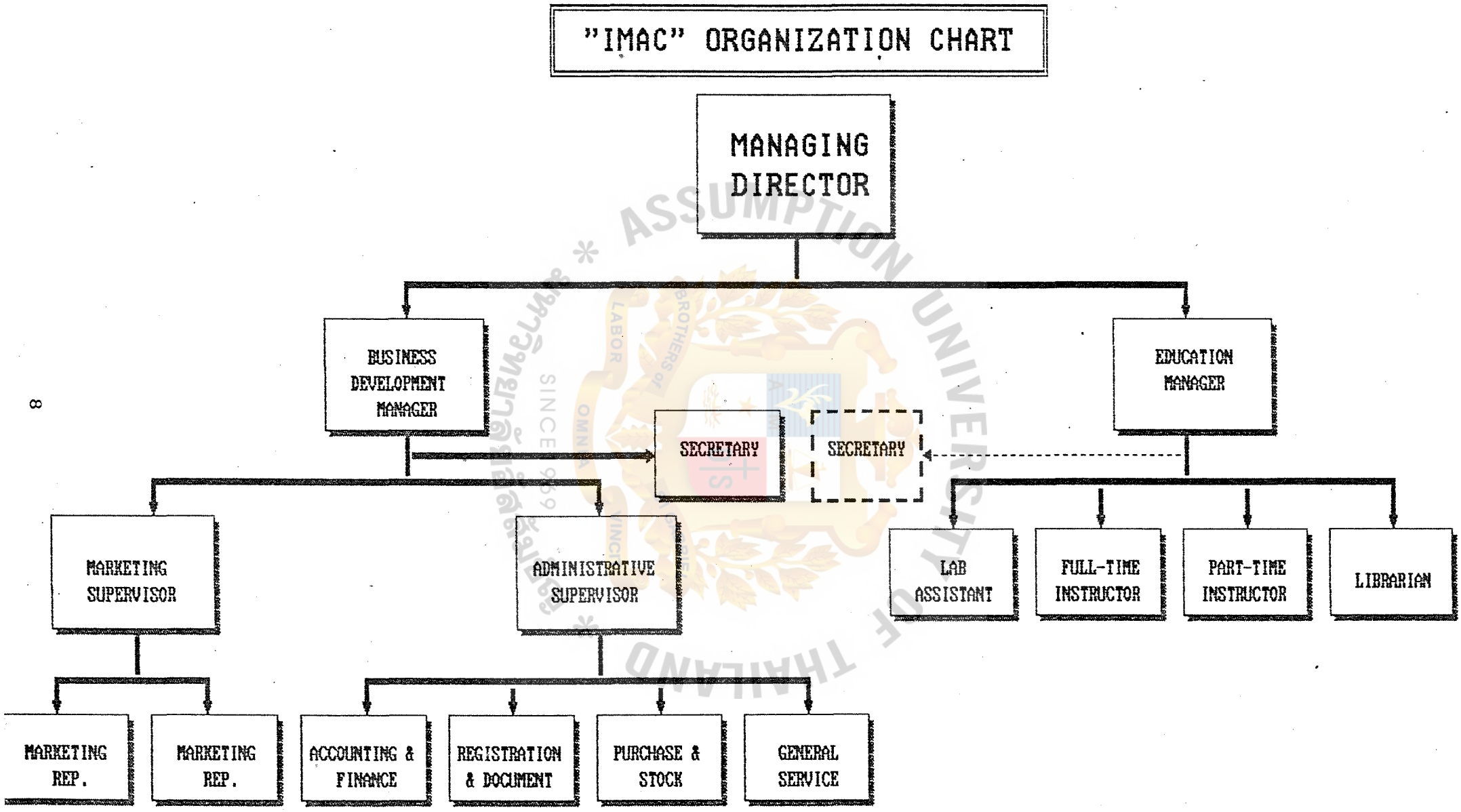


FIGURE 2-1 ORGANIZATION CHART OF "IMAC"

## 2.3 CURRENT PROBLEMS

A problem is a deviation between a desire and an actual or anticipanted situation. The main reasons for wanting to improve the present situation by developing the new system are :

1. Lack of integration between existing system.

2. Inability of the organization to make optimum use of information that is available.

3. Need better access to existing system, for example, student registration, cost control information.

4. Inability to easily query existing data & provide management level reporting.

5. A changing business environment that is not addressed by the current system.

6. Inadequate use of new computer technology and software.

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

All these problems lead to :

- User dissatisfactions
- Gaps in data flow
- Late and inadequate reports
- High level of maintenance on existing system

## 2.4 AREA OF IMPROVEMENT

There are many possibilities for improvement which include :

(a) Improving integration of systems and subsystems.

(b) Improving staff satisfaction with the system.

(c) Improving ease of student/prospect interaction with the system.

(d) Reducing redundant output.

(e) Elimination of unnecessary or duplicated steps.

(f) Improving speeding up a process.

ABAC
GRADUATE SCHOOL LIBRARY

111 c.1

# CHAPTER 3

# PROPOSED SYSTEM

## 3.1 USER REQUIREMENTS

IMAC needs to achieve greater student registration efficiency. The current procedures have been plagued with difficulties for years. There is a strong demand for an on-line system to provide increased currency in reporting class roster and other information. Grade reporting and transcript preparations should also be integrated into the system.

Instructors also need a computerized system in performing services that are inconvenient to provide with the present system. Currently, instructors are responsible for determining students' grades. It is hoped that the new system will be able to automate grade checking system, too.

Another requirement for IMAC is to improve the cost control procedure and management information system for management decision-making.

## 3.2 OVERVIEW OF THE PROPOSED SYSTEM

The new computerized information system will cover the following aspects :

The scope of the proposed system is defined by the context diagram shown in Figure B-1. As indicated, the computerized information system interact with five external entities.

- The registration office will use the new system for on-line student master information. The on-line registration capability was one of the prime motivations for authorizing development of the new system. The processing of registration request is the preparation of student course schedule. Students will also receive transcripts or certificates upon submitting requests to the registration office.

- An instructor receives a class roster from registration office. At the end of the term, he/she will submit student grades to the registration office for inputing into the student grade system.

- The financial department issues a receipt after processing the student payment system. The detail amount that the student must pay will be shown on the screen. So, the student can recheck with his/her registration form. The department also calculates cost control information for each course especially for the short course. This is the valuable description for the company to determine course-offering for each semester.

- Management ( Business development manager and Education manager) are responsible for set up class planning. They also involve instructors to plan the suitable course of each semester.

- Prospect is the last external entity mentioned in this development system. Since IMAC is a private school, prospect information is an important marketing strategy for management. Performance evaluation from prospect is an exploited information to monitor student progress, too.

From the context diagram in Figure B-1. A set of data flow diagram level segments are developed into a diagram 0 that described the completely integrated system as shown in Figure B-2. The DFD indentifies the major processing events in the system and show the data flows and files that interact to transform the inputs into outputs for each process. The eight events of this proposed system are :

1. Build class file
2. Build plan table file
3. Build master file
4. Process registration
5. Process payment
6. Prepare class schedule
7. Process grade
8. Monitor student progress

Process 1 : Build class file

1.1 The class cost control information is submitted by the financial dept. for management for use as a criteria in opening each class and for building the class cost control file. Class planning is determined by the management for the current term.

1.2 The student class file is built with information from the class planning and the class cost control file.

Process 2 : Build plan table file

2.1 The plan table file is built with information from the management level.

Process 3 : Build master file

3.1 An application form is submitted by the student. It is checked by the registrar's office. The student master file is built with information from this application form.

3.2 An instructor list is sent to registrar's office for enter/update instructor information to the instructor master file.

3.3 The prospect master file is built from outside information, such as printing materials, medias. etc.

Process 4 : Process registration

4.1 Student fills-in a registration request to the registrar's office.

4.2 The process registration is carried out on-line. Course requests are entered interactively into the system. The student class file is accessed to verify the course description and check on prerequisites.

4.3 After the final schedule is verified. A record is written to the student transaction file and student summary file.

4.4 A registration form is prepared for the student.

Process 5 : Process payment

5.1 Student brings the completed registration form for payment to the financial department.

5.2 The payment system is carried out on-line. The student transaction file and student summary file are accessed to display all student registration records.

5.3 The payment can be made in cash or check. After the payment record is verified. The completed record is added to the student summary file as necessary.

5.4 The student payment report is prepared to the financial department.

Process 6 : Prepare class schedule

6.1 The student class file, student transaction file, instructor file, plan table file, student master file, are accessed for class schedule.

6.2 For each course, the student class file is accessed by course number + section number. The student transaction file is accessed by student id. The instructor master file & student master file are accessed to get names for printing on the class roster.

6.3 The course schedule for students is listed from the student transaction file.

Process 7 : Process grade

7.1 Student grades are submitted by each instructor and are listed sequence by student id.

7.2 The student grade system is used to process grades by registrar's office. The scores, grades, GPA values are updated into the student transaction file. Cumulative GPA are also updated in student master file.

7.3 The student's certificate and student's transcript are prepared from the student transaction file. The transcript is printed in order by term.

Process 8 : Monitor student progress

8.1 Prospect master file is listed for job requisition. Training jobs for student will be provided by prospect. Any performance evaluation will be a record written to the student master file.

8.2 Information from the student master file will be used to prepare student progress reports to management.

The files used in this proposed system are :

1. Student master file
2. Prospect master file
3. Instructor master file
4. Plan table file
5. Student class file
6. Class cost control file
7. Student transaction file
8. Control running file
9. Student summary file

Inquiry and hard-copy reports are designed to meet user requirements They consist of :

1. For top and middle management levels

   Inquiry :

   - Student information by student id.
   - Student information by student name
   - Course schedule by plan no.

   Hard-copy reports

   - Student information by media type
   - Top ranking analysis by area code
   - Class-offering report
   - Cost control analysis report
   - Student outlet by course no.
   - Student registration in each term
   - Tuition fee for each course
   - Student movement information in each year
   - Student progress summary report

2. For operational level

   Hard-copy reports

   - Student name & address run book
   - Prospect information run book
   - Student payment report
   - Student registration in each term

3. For other requirements

   Forms and reports

   - Registration form
   - Receipt
   - Course schedule for student
   - Grade report for student
   - Student's transcript as request
   - Student's certificate as request

- Class roster for instructor
- Student grade report for instuctor

## 3.3 HARDWARE REQUIREMENTS

The hardware of this proposed system can be configured as follows :

| | QTY | BAHT |
|---|---|---|
| 1. File Server | 1 | 93,000 |
| - CPU Intel 80486 | | |
| - 4 MB RAM | | |
| - 200 MB Hard Disk | | |
| 2. Work Stations | 3 | 150,000 |
| - IBM PS/1 Pro Model 086 or IBM PC AT Compatible | | |
| - 2 MB RAM (Maximum on system board 16 MB) | | |
| - 386SX 20 MHZ. | | |
| - 1.44 MB 3.5" Diskette Drive | | |
| - Keyboard | | |
| - 14" VGA Color Display | | |
| 3. Printer | 2 | 46,000 |
| - Epson Printer LQ1170 | | |
| - Printer Cable | | |
| 4. Ethernet Cards for Server and work stations | 4 | 28,000 |
| 5. Cabling | | 3,000 |

The above hardware can be purchased from outside vendors.

## 3.4 SOFTWARE REQUIREMENTS

The software to be used by this system are :

- MS DOS Version 5.0
- "Genifer" Application generator (the original Genifer package contains templates for menu program, file maintenance program, inquiry and report program)
- DBASE III PLUS or FoxBASE
- Computer system programs for IMAC
- Manual documentation

Other requirements for this proposed system are :

- 5.25" Floppy disks (optional)
- 9x11, 11x11, 15x11 continuous forms
- Preprinted forms (registration form, receipt form, etc.)
- Computer desk and printer stand (optinal)

## 3.5 SYSTEM REQUIREMENTS

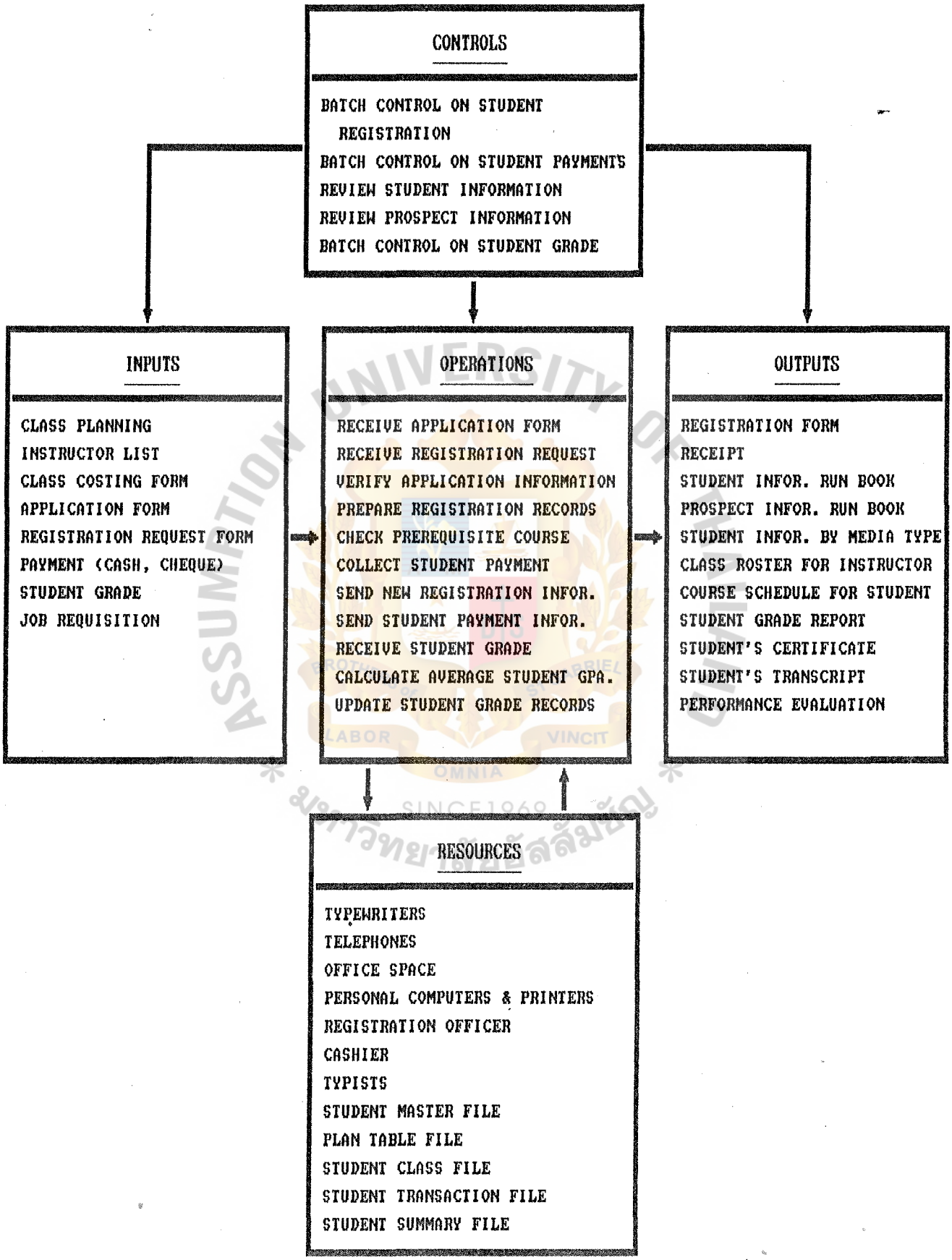


FIGURE 3-1 SYSTEM REQUIREMENTS MODEL FOR IMAC'S COMPUTERIZED INFORMATION SYSTEM

## 3.6 COST & BENEFIT ANALYSIS

The method for evaluating the proposed computer information system is the cost/benefit analysis. When computer information systems are concerned, costs can be difficult to calculate. Benefits can be even harder to determine than costs. For this company, costs and benefits are studied and quantified separately. It encounter 2 types of costs and benefits: Tangible and Intangible [6].

1. Tangilble costs - stated in term of the purchase price or payout over the useful life of the system. They are often converted to operational terms.

- Computer equipment
- Personnel cost
- File conversion cost
- Software cost
- Facilities (power, space, etc.)

2. Tangible benefits - are realized when the proposed system is projected to make or save money of its organization.

- Reduction in payroll cost
- Faster processing of operations

3. Intangible costs - are those that can't be easily pinpointed in term of money.

- Higher error rates during conversion phase
- Decreased output from employee because of apprehension the changes in job content

4. Intangible benefits - are those delivered, indefinable improvement for which values can be elusive.

- Improve decision process by providing faster access to information
- Improvements in customer services
- Improvements in employee morale
- Smoothing of operational flows

After costs and benefits have been identified and their value estimated, several methods can be used to support decisions on the financial feasibility of proposed system. The Payback method is the technique applied to cost /benefit analysis of this computerized information system. A straight forward method for determining payback is to apply the cost saving anticipanted each year against outstanding investment cost [1].

New System Cost :

| | BAHT |
|---|---|
| Hardware | 320,000 |
| Software | 80,000 |
| Personnel | 60,000 |
| Training | 10,000 |
| Supplied | 3,000 |
| Overhead | 7,000 |
| Total Investment Cost | 480,000 |

| YEAR | COST BAHT | CUMULATIVE COSTS BAHT | BENEFITS BAHT | CUMULATIVE BENEFITS BAHT |
|---|---|---|---|---|
| 0 | 480,000 | 480,000 | 0 | 0 |
| 1 | 120,000 | 600,000 | 250,000 | 250,000 |
| 2 | 100,000 | 700,000 | 370,000 | 620,000 |
| 3 | 100,000 | 800,000 | 450,000 | 1,070,000 |
| 4 | 120,000 | 920,000 | 520,000 | 1,590,000 |
| 5 | 140,000 | 1,060,000 | 650,000 | 2,240,000 |

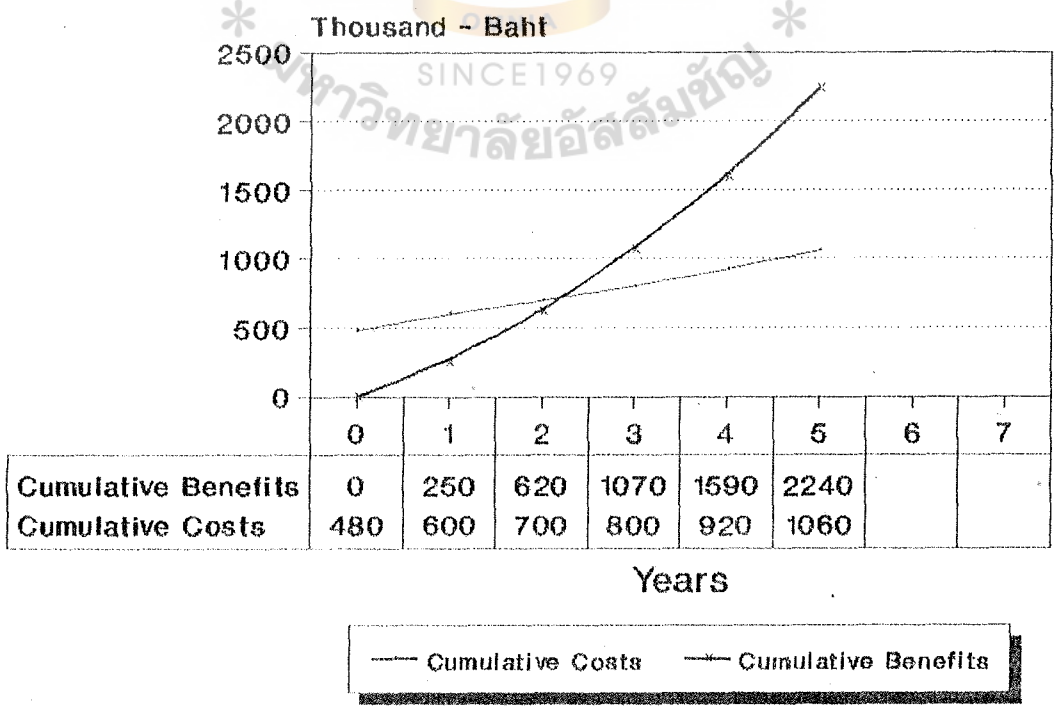


Figure 3-2 Payback analysis showing a payback period

# CHAPTER 4

# IMPLEMENTATION PLAN

The activities associated with implementing the completed system are : establishing the interface with the existing applications, programming, testing, user training, and finalizing documentation.

## 4.1 PROGRAMMING

IMAC computerized information system is developed through prototyping with dBASE III PLUS and supplemented by GENIFER. Programs are developed according to the following processes :

(a) Problem analysis

(b) Program design

(c) Program specifications

(d) Program coding

(e) Program debugging & testing

The following figure shows the hierarchy of the system menu.

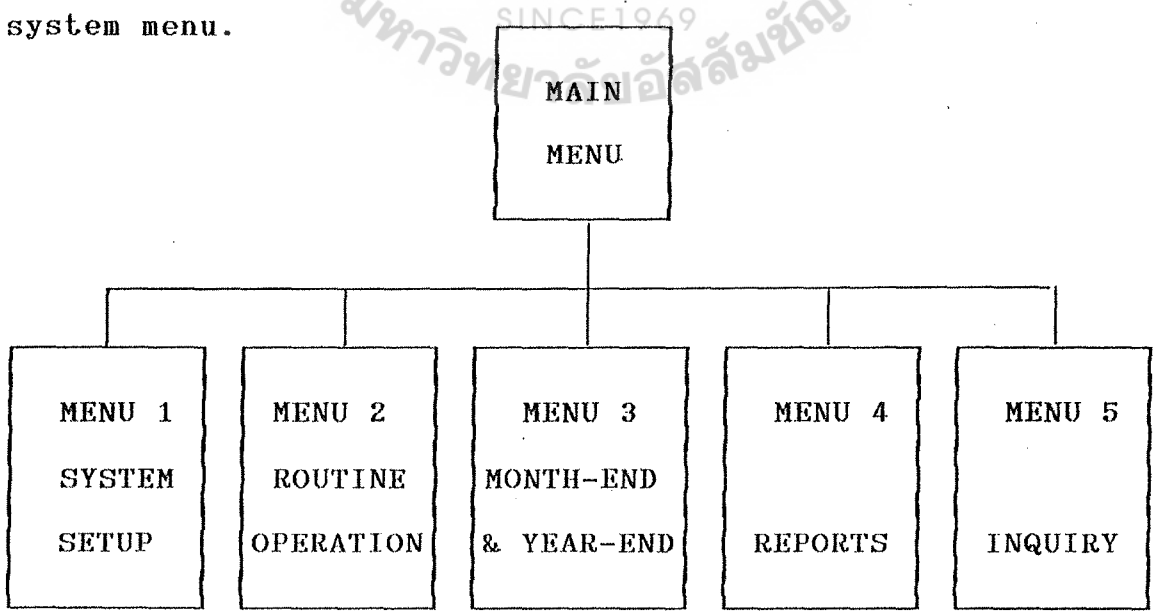


Figure 4-1 The hierarchy of IMAC Information System

Program Package for IMAC :

System Setup :

ST001 - Enter/Update Student Master Information

ST002 - Enter/Update Student Class Information

ST003 - Enter/Update Instructor Master Information

ST005 - Enter/Update Plan Table Information

ST010 - Enter/Update Prospect Master Information

Routine Operation :

ST004 - Maintain Class Cost Control Analysis

ST007 - Process Student Registration as Request

ST006 - Process Student Payment as Request

ST008 - Process Student Grade as Request

Month-end and Year-end Works :

ST009 - Post Student History File

ST012 - Chop Student Completed Record

Reports and Inquiry :

ST101 - Student Name and Address Run Book

ST102 - Student Information by Media Type

ST103 - Class-Offering Report

ST104 - Cost Control Analysis Report

ST105 - Student Outlet by Course No.

ST106 - Student Registration in Each Term

ST107 - Tuition Fee for Each Course

ST108 - Student Movement Infor. in Each Year

ST109 - Student Progress Summary Report

ST110 - Top Ranking Analysis by Area Code

ST111 - Prospect Information Run Book

ST112 - Student Payment Report

ST121 - Course Schedule Listing for Student

ST122 - Grade Report for Student

ST123 - Student's Transcript as Request

ST124 - Student's Certificate as Request

ST125 - Class Roster for Instructor

ST126 - Student Grade Report for Instructor

Inquiry :

ST201 - Student Information by Student ID

ST202 - Student Information by Student Name

ST203 - Course Schedule by Plan No.

4.2 TESTING

All new application programs must be tested thoroughly. Testing of IMAC computerized information system is done throughout the development process which are divided as follows:

(a) Hardware testing

1. Normally, this is provided by the equipment vendor upon its delivery.

(b) Software testing

It consists of :

1. Test programs to verify the way the system works. For IMAC computerized Information System, It will start with set-up programs, routine or operation programs, report and

inquiry programs. For instance, the registration program should be able to verify that each student does not register more than 4 courses in one semester or prerequisite courses have been taken.

2. The data testing is done by creating both valid and invalid data. This created test data must have minimum and maximum value, as well as all possible variations in format and codes.

   An example of data testing in this proposed system can be:

   2.1 Create test data in the student class file, the plan table file, and the student master file.

   2.2 Call the student registration program for data entry. Check the 'total fees' that must match the result from adding the amounts in the student class file and the plan table file.

3. System testing with live data is completed when the testing with the test data proves to be satisfactory. After this, the output should be finally checked.

## 4.3 USER TRAINING

The purpose of the user training is to prepare users for the ongoing operation of the system. The

objective of this training is to make the user managers and the operating staff understand the proposed system. The scope of the training effort will encompass all personnel in IMAC who will be affected by the new system.

Training for IMAC Computerized Information System will cover menu-driven report and inquiry for all operational, middle and top managements, file maintenance and program usage for registration and finance officers.

### 4.4 FINALIZING DOCUMENTATION

Documentation serves both the user and management groups of the organization [2]. The documentation of IMAC includes :

1. Operator instructions.
2. Identification of responsible personnel.
3. Database design (Appendix C).
4. Program and file specifications (Appendix F).
5. System analysis and design models (Appendices B, D, E, and G).
6. Report to management (Appendix H).

# CHAPTER 5

# CONCLUSIONS & RECOMMENDATIONS

## 5.1 CONCLUSIONS

The summary of the whole system development project is as follows :

The entire study was conducted to improve the existing student registration, payment, and student grade system which were carried out manually. The need of a computerized system is in providing data and information relating to an organization's operations. This project supports the activities of employees, owners, customers, and other key people in the organization, either by effectively processing data to assist with the transaction work load or by effectively supplying information to the management in timely manner [5].

The proposed system consists of the following subsystems.

1. Transaction processing systems - comprises of the routine processing functions :
   - Student registration system
   - Student payment system
   - Student grade system
2. Information systems - comprises of the on-line inquiry and printed reports.

- Class roster
- Course schedule
- Student registration in each term
- Student information in each year
- Student payment
- Student progress report

3. Decision support systems - provides a set of easy-to-use models, retrieving and reporting capabilities so management can generate the information they feel will be useful to them in making decisions [3].
   - Class cost control analysis for planning
   - Prospect information and student information which to be helpful in marketing strategy.
   - Query information as management request

The software component of the IMAC computerized information system was developed through prototyping with dBASE III Plus supplemented by GENIFER.

The output generations can be either the on-line inquiry or hard-copy reports.

The cost/benefit analysis proves the feasibilities of the proposed system in term of the pay-back period. This approach attempts to measure the time it will take to recoup in benefits the original money invested [4].

The implementation phases are : programming, testing, user training, finalizing documentation.

Of course, the new system does not provide all possible services to the identified entities or to other potential users of the system but it represents a system that can be expanded later with additional features.

## 5.2 RECOMMENDATIONS

After the new system is installed for a period of time and users become familiar with the system, it is recommended that the guidelines given below be followed :

5.2.1 For IMAC to be fully computerized in the near future, other systems should also be considered; for example, Accounting Information System (together with its subsystems i.e., accounts payable, accounts receivable, account and finance systems), Library System for use in the institute's library, etc.

5.2.2 Expand a better information service by designing a user-friendly interface that allows the students to directly access the information they need; for example, courses opened for the semester, total number of students accepted for a course and number

of students already registered, etc., from computers which are installed in the provided areas.

5.2.3 Redesign a more sophisticated control and security procedure by :

(a) Providing protection against unauthorized use and invasion of privacy of specified files.

(b) Backing up data frequently and keeping copies of data files in a safe palce or even in an offsite location.

(c) Defining duties, responsibilities and access restrictions for all personnel concerned.

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

# REFERENCES


1. Adams, D. R., M. J. Powers, and V. A. Owles, Computer Information Systems Development : Design & Implementation, South-Western Publishing, 1985.

2. Amadio, W., and C. Amadio, System Development Projects, McGraw-Hill Inc., 1990.

3. FitzGerald, J., and A. F. FitzGerald, Fundamentals of System Analysis (3rd Edition), John Wiley & Sons, Inc., 1987.

4. Kendall, K.E., and J. E. Kendall, System Analysis & Design, Prentice-Hall, 1988.

5. Parker, C. S., Management Information Systems : Strategy and Action, McGraw-Hill Inc., 1989.

6. Powers, M. J., D. R. Adams, and H. D. Mills, Computer Information Systems Development : Analysis & Design, South-Western Publishing, 1984.


ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

# APPENDICES

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

# APPENDIX A
# EXISTING FORMS

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST. GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

RAINTREE OFFICE GARDEN
A3 BUILDING, 4th FLOOR
272 SOI JAPANESE SCHOOL,
RAMA IX ROAD, HUAY KWANG,
BANGKOK 10310. THAILAND.
TEL. (02) 3199950-3 FAX. 3199954

Application No.

Application Date(D) (M) (Y)

Course Applied

Year Semester Shift

Name (E)

Marrital Status S = Single, M = Married, O = Other

Sex M = Male, F = Female

Date of Birth (D) (M) (Y)

Nationality ..... Religion .....

Present Address

Post Code Telephone .....

Permanent Address

.....

.....

Occupation Title

Office Address

.....

.....

Figure A-1 Application Form of the Existing System.



TECHNOLOGY SUPPLY GROUP
IBM Business Partner CMAP

Academic Background :

| Schools/Universities Attended | Date Attended | | Certificate/Diploma/Degree and major field |
|---|---|---|---|
| | From | To | |
| | | | |
| | | | |
| | | | |

I certified that the information given in this application form is complete and accurate to the best of my knowledge. If admitted, I agree to comply with the education regulations of the Information Management Academic Centre.

Signature .....

Date .....

OFFICIAL USE ONLY

Student I.D. ☐☐ - ☐☐☐ - ☐ - ☐☐☐

Document Received:

- ☐ 1.5 inch photographs
- ☐ Photocopy of ID Card
- ☐ Photocopy of Domicile Declaration
- ☐ educational references ( Transcript/Certificate/Degree )

Officer Name .....

Figure A-2 Application Form (Cont'd) of the Existing System.

RAINTREE OFFICE GARDEN
A3 BUILDING, 4th FLOOR
272 SOI JAPANESE SCHOOL,
RAMA IX ROAD, HUAY KWANG,
BANGKOK 10310, THAILAND.
TEL. (02) 3199950-3 FAX. 3199954

*TRAINING REGISTRATION*

COURSE TITLE _____

Kindly complete and return to : Applied Technology Co.,Ltd. (IMAC)
Raintree Office Garden
A3 Bld. 4th Fl.
272 Soi Japanese School
Rama IX Rd. Huay Kwang
Bangkok 10310

Company _____

Address _____

_____

_____

Telephone _____ Fax. _____

| Name of participants | Position |
|---|---|
| 1. _____ | _____ |
| 2. _____ | _____ |
| 3. _____ | _____ |

Fees enclosed Baht/person _____

Method of payments:

( ) Check enclosed : Payble to : APPLIED TECHNOLOGY CO.,LTD.

Check No. _____ Amount _____ B

Bank _____ Branch _____

Date _____

( ) Cash _____ B (_____)

_____

(Authorized Signature)

Figure A-3 Registration Form of the Existing System.

For more information: please contact IMAC

Tel. 319-9950-1 Fax. 319-9954



TECHNOLOGY SUPPLY GROUP

IBM Business Partner CMAP

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST. GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

TS TECHNOLOGY SUPPLY GROUP

IBM Business Partner CMAP

# ตัวอย่างใบเสร็จรับเงิน

No: X010046

INFORMATION MANAGEMENT ACADEMIC CENTRE
APPLIED TECHNOLOGY Co., Ltd.
MAINTREE OFFICE GARDEN, A3 BUILDING 4TH FLOOR
27/1 SOI JAPANESE SCHOOL, RAMA IX ROAD, HUAY KWANG
TEL. 3199950-3 FAX. (02) 3199954

RECEIPT

DATE / /

RECEIVED FROM :
ADDRESS :
IN PAYMENT OF :

BAHT

~~CASH~~ /CHEQUE NO. BANK

BRANCH DATE

IF PAYMENT IS MADE BY CHEQUE, THIS RECEIPT IS VALID SUBJECT TO REALIZATION OF THE CHEQUE

COLLECTOR DATE / /

Figure A-4 Receipt Form of the Existing System.

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE 1969
มหาวิทยาลัยอัสสัมชัญ

I M A C

ACADEMIC RECORD

INTERNATIONAL DIPLOMA IN

COMPUTER STUDIES

NAME _____

STUDENT ID. _____

TERM _____

| Course No. | Course Title | Credit | Grade |
|---|---|---|---|
| 101 - 201 | Computer Concept II | 3 | |
| 101 - 205 | Information Management | 3 | |
| 101 - 207 | COBOL Programming | 3 | |
| 101 - 307 | Computer at Works | 3 | |

Grade Code : A = 90% up , A- = 86% - 89% , B+ = 80% - 85% , B = 75% - 79%
B- = 70% - 74% , C+ = 65% - 69% , C = 60% - 64% ,
D = 50% - 59%

_____ _____

Date of Issue Education Manager

Figure A-5 Grade Report of the Existing System.

แบบประเมินผลการฝึกงานของนักศึกษา

หลักสูตร ประกาศนียบัตรคอมพิวเตอร์ขั้นต้น

(Certificate in Computer Programming)

สถาบัน IMAC (INFORMATION MANAGEMENT ACADEMIC CENTRE)

ชื่อนักศึกษา__นายสุชาติ_____นามสกุล__อภิรักษ์สุขมาล_____

สถานที่ฝึกงาน___บริษัท_บี._เอ็ม._คอมพิวเตอร์__จำกัด_____

ตั้งแต่วันที่__1___เดือน_กันยายน_____พ.ศ._2535__ถึงวันที่__24__เดือน_ตุลาคม_____พ.ศ._2535__

โปรดกาเครื่องหมาย / ลงในช่อง ☐ ที่ตรงกับข้อความที่ท่านต้องการ และเติมข้อความหรือตัวเลขลงช่องว่างที่กำหนดให้

| | ดีมาก | ดี | ปานกลาง | ต้องปรับปรุง | ใช้ไม่ได้ | ประเมินไม่ได้ |
|---|---|---|---|---|---|---|
| 1. ความรู้เกี่ยวกับงานที่ได้รับมอบหมาย | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |
| 2. ความสามารถในการใช้เครื่องมืออุปกรณ์ | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |
| 3. ความสนใจในงาน | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |
| 4. ความสามารถในการเรียนรู้ | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |
| 5. ความคิดริเริ่ม | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |
| 6. การรับฟังความคิดเห็นของผู้อื่น | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |
| 7. การทำงานร่วมกับผู้อื่น/มนุษยสัมพันธ์ | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |
| 8. ความรับผิดชอบต่องาน | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |
| 9. คุณภาพของงาน | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |
| 10. การตรงต่อเวลาในการทำงาน | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |
| 11. การอยู่ปฏิบัติงานอย่างสม่ำเสมอ | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |
| 12. กิริยามารยาท | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |
| 13. ความเรียบร้อยของการแต่งกาย | ☐ | ☐ | ☐ | ☐ | ☐ | ☐ |

Figure A-6 Evaluation Form of the Existing System.



(โปรดพลิก)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE 1969
มหาวิทยาลัยอัสสัมชัญ

ความคิดเห็นอื่น ๆ / ข้อเสนอแนะ

---

---

---

---

บันทึกการลา / มาสาย

ลากิจ _____ วัน ลาป่วย _____ วัน

มาสาย _____ วัน อื่น ๆ _____ วัน

รวม _____ วัน

ผลการประเมินรวม

เมื่อพิจารณาทั้งการประเมินผลการฝึกงานและการลา/มาสาย สมควรให้

☐ ผ่านการฝึกงาน ☐ ไม่ผ่านการฝึกงาน

การรับเข้าทำงาน

หลังจากการฝึกงานเสร็จสิ้นลงแล้ว บริษัทของท่านต้องการรับ นายสุชาติ อภิรักษ์สุขุมาล เข้าเป็นพนักงานหรือไม่

☐ รับ ☐ ไม่รับ

เพราะ_____

---

ลงชื่อ _____

( )

ตำแหน่ง _____

ผู้ประเมินผลการฝึกงาน

วันที่ _____ เดือน _____ พ.ศ. _____

Figure A-7 Evaluation Form (Cont'd) of the Existing System.

# APPENDIX B
# CONTEXT AND
# DATA FLOW DIAGRAMS

ASSUMPTION UNIVERSITY OF THAILAND
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

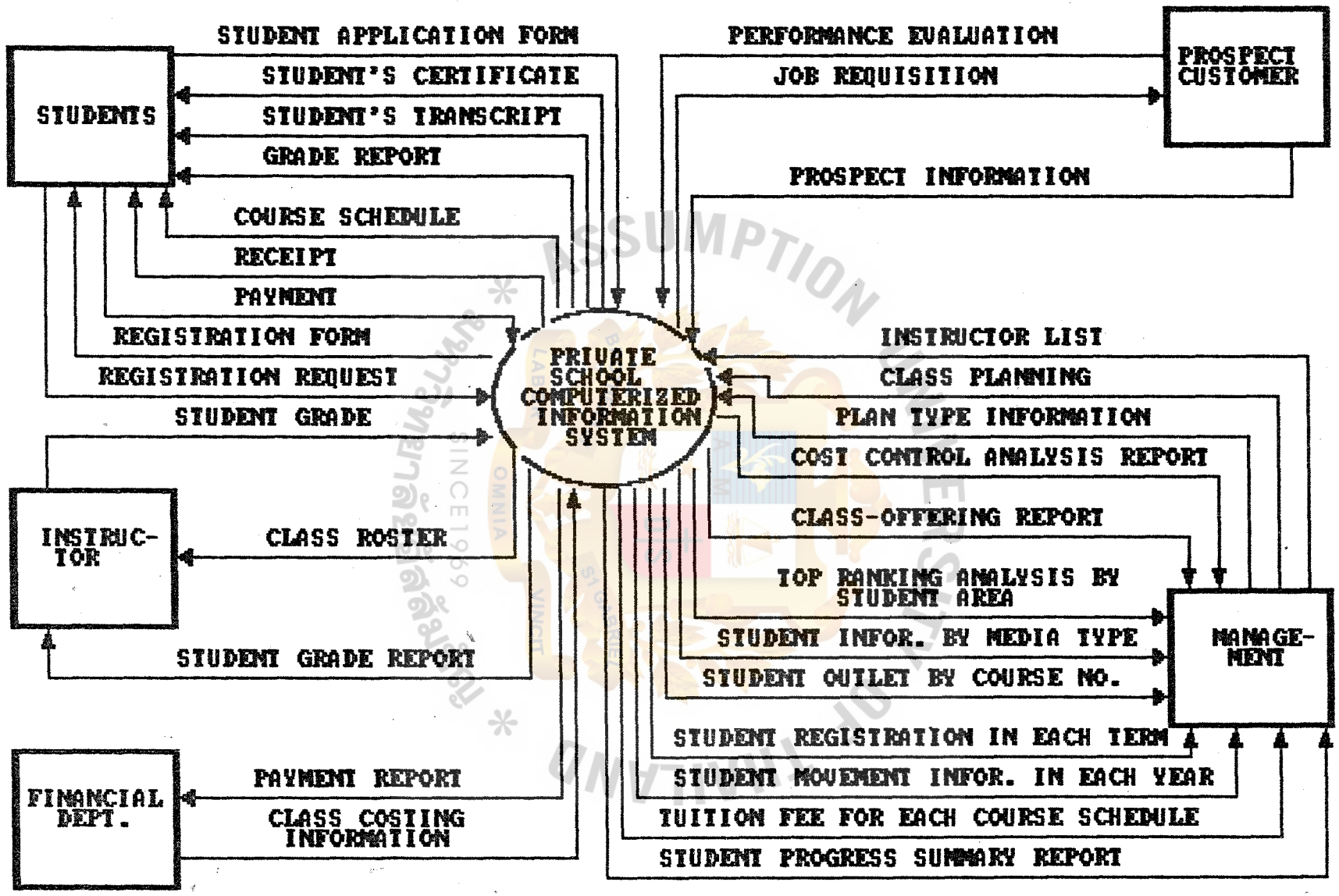


FIGURE B-1 CONTEXT DIAGRAM FOR PRIVATE SCHOOL COMPUTERIZED INFORMATION SYSTEM

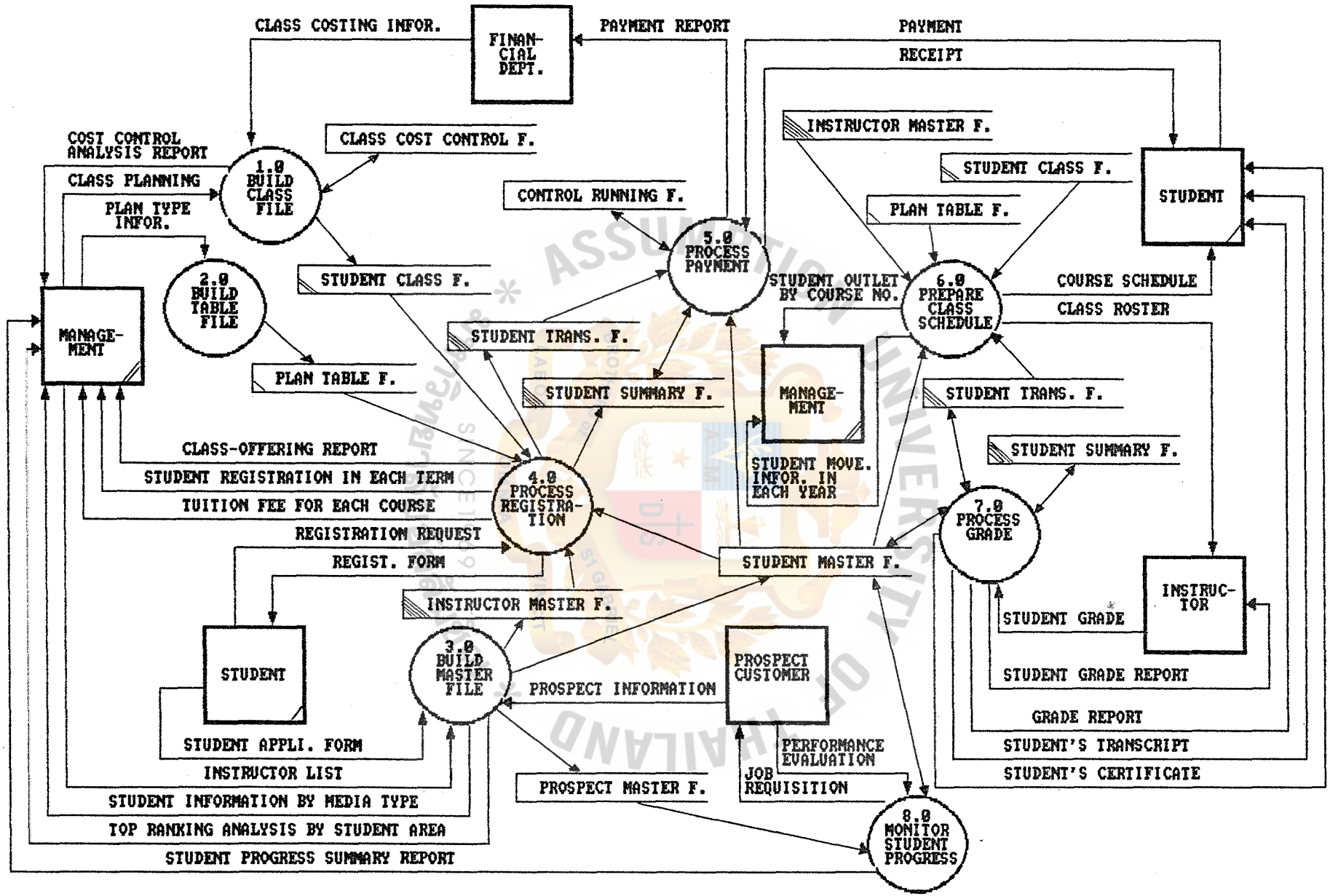


FIGURE B-2 DFD LEVEL 0 (A DERIVED MODEL OF "PRIVATE SCHOOL COMPUTERIZED INFORMATION SYSTEM" )

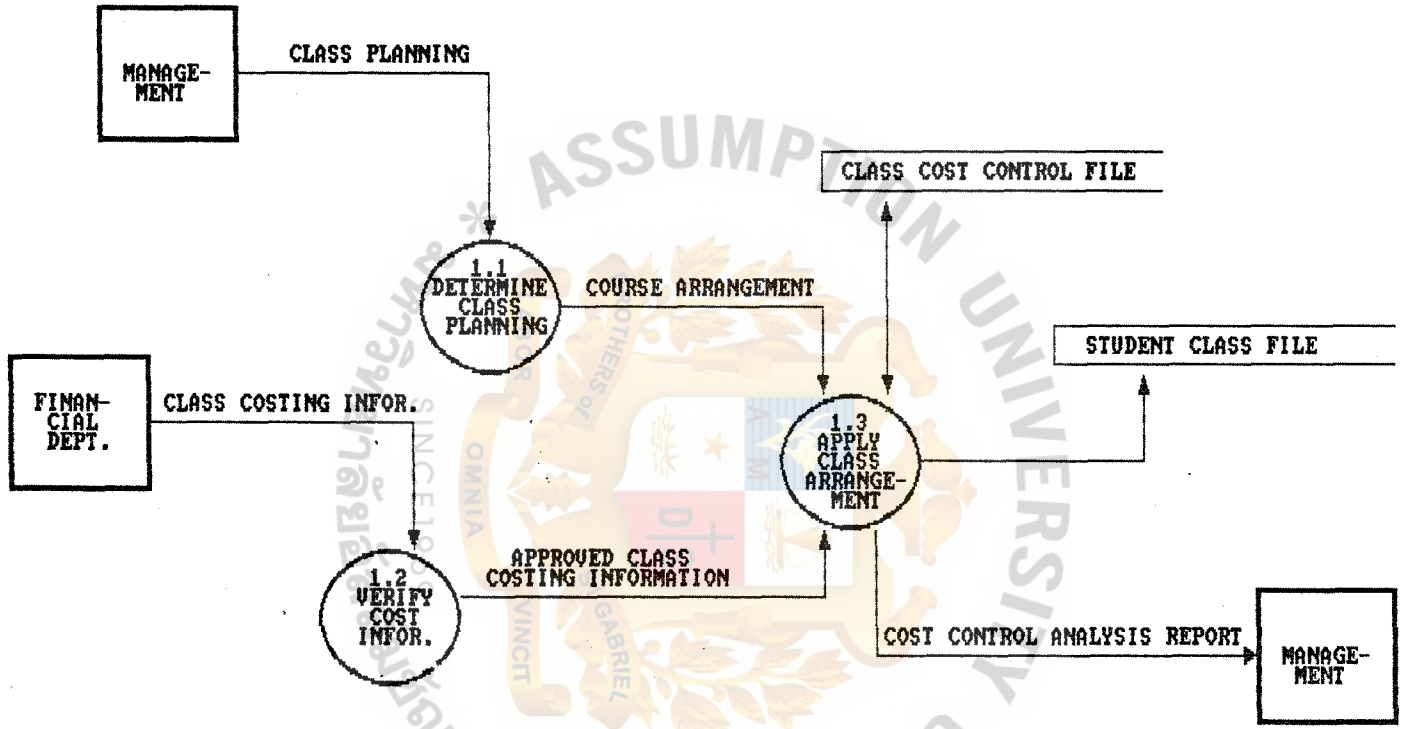


FIGURE B-3 DATA FLOW DIAGRAM LEVEL 1. PROCESS 1 : BUILD CLASS FILE

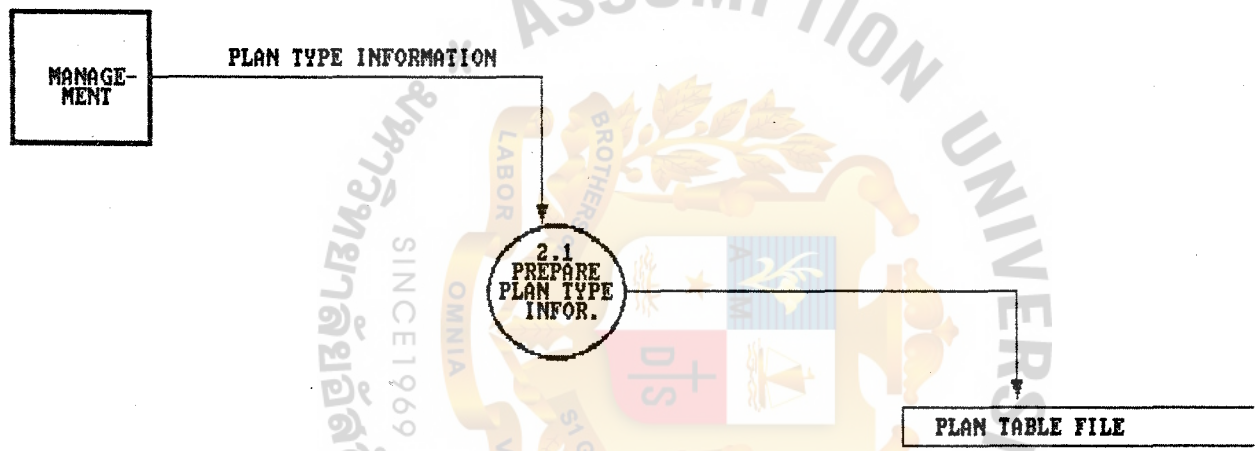


FIGURE B-4 DATA FLOW DIAGRAM LEVEL 1. PROCESS 2 : BUILD PLAN TABLE FILE

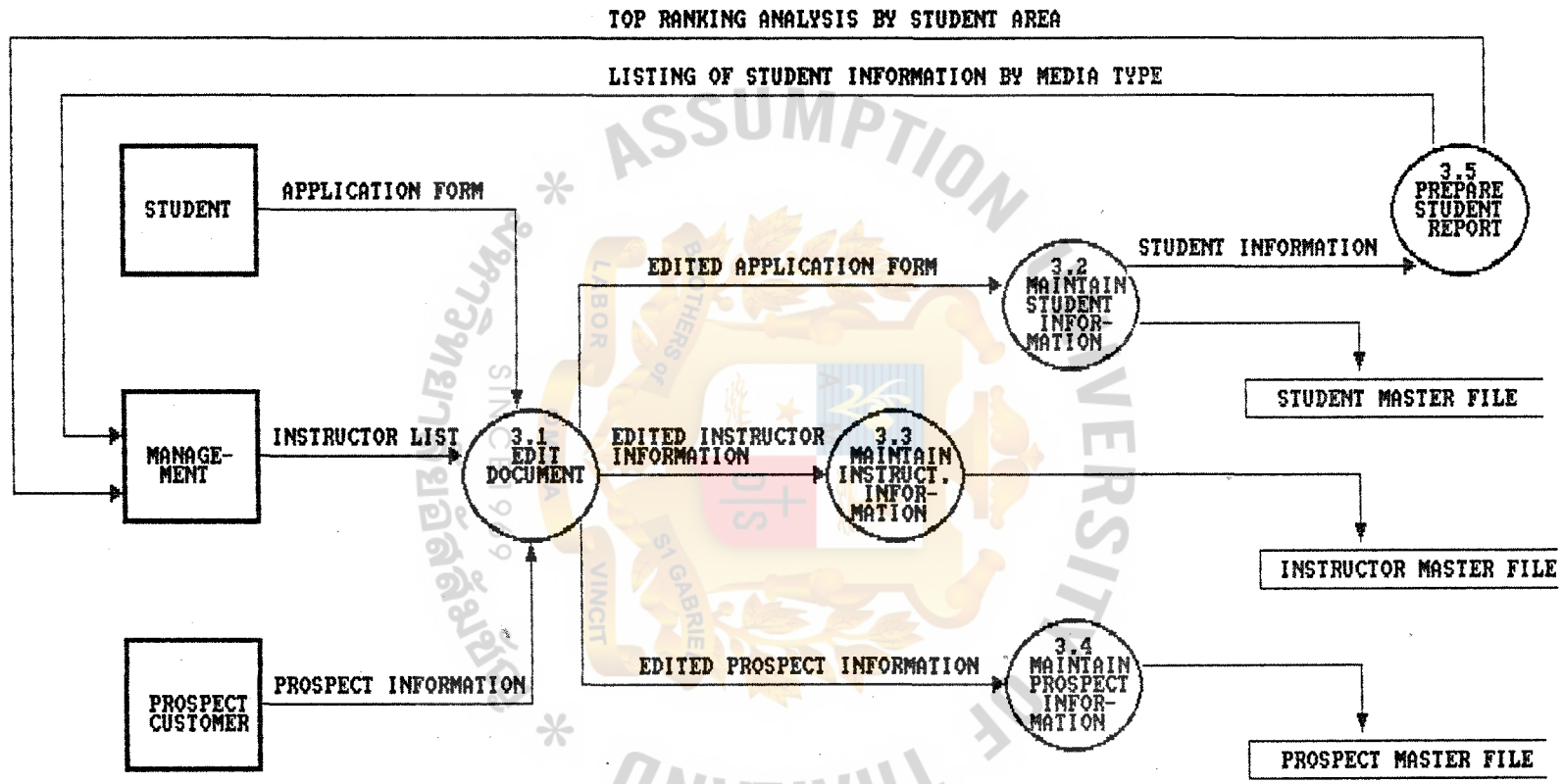


FIGURE B-5 DATA FLOW DIAGRAM LEVEL 1. PROCESS 3 : BUILD MASTER FILE

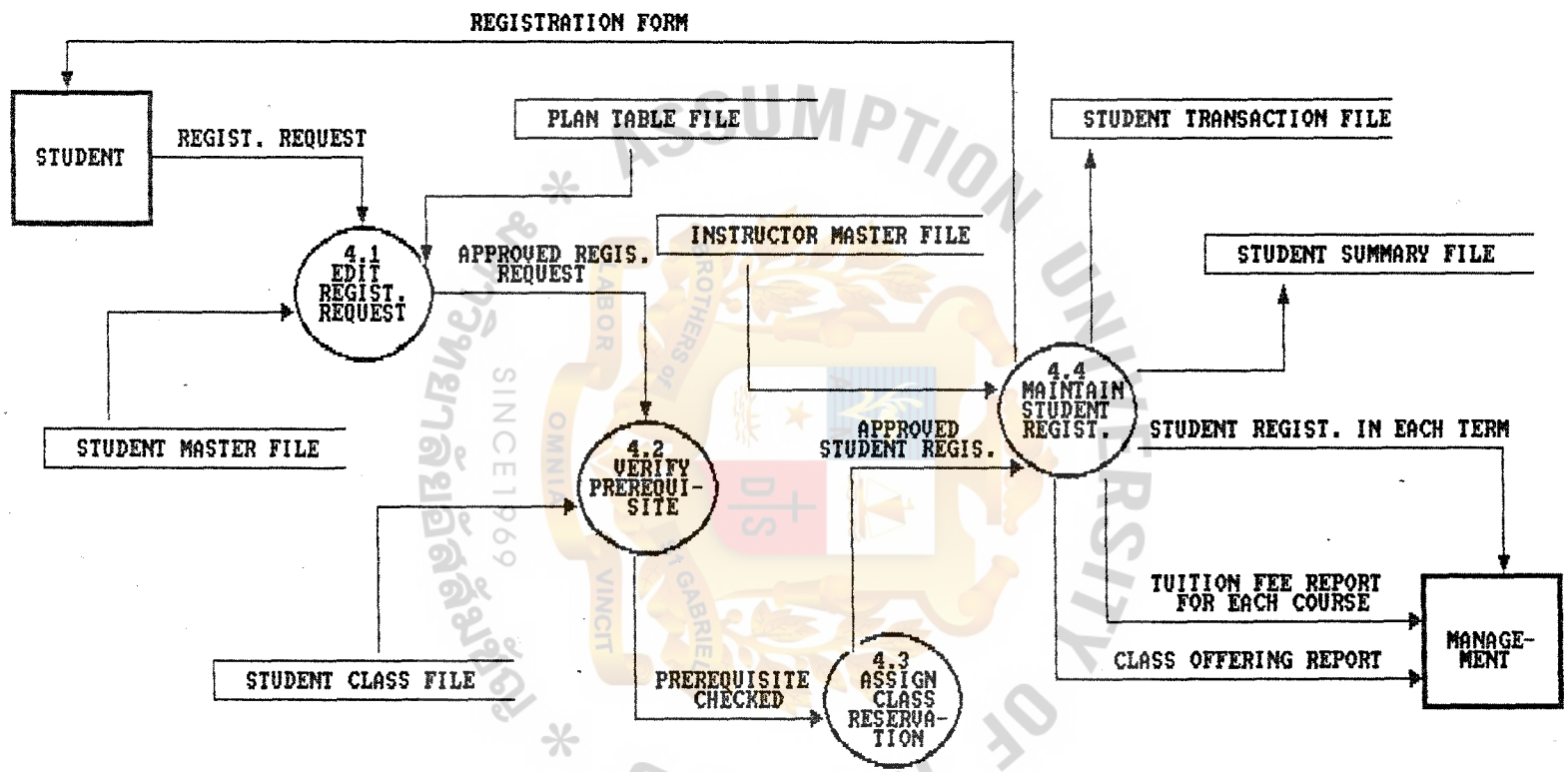


FIGURE B-6 DATA FLOW DIAGRAM LEVEL 1. PROCESS 4 : PROCESS REGISTRATION

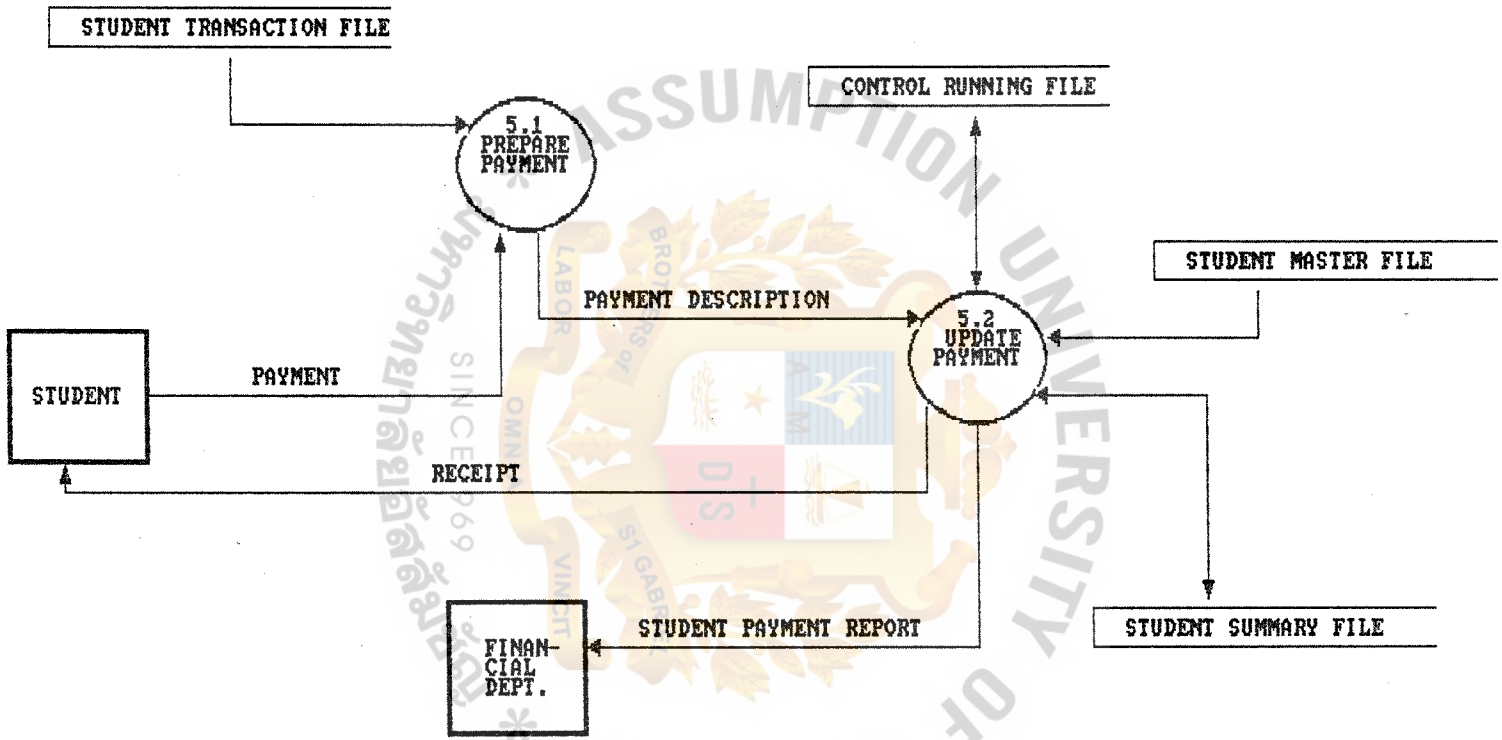


FIGURE B-7 DATA FLOW DIAGRAM LEVEL 1. PROCESS 5 : PROCESS PAYMENT

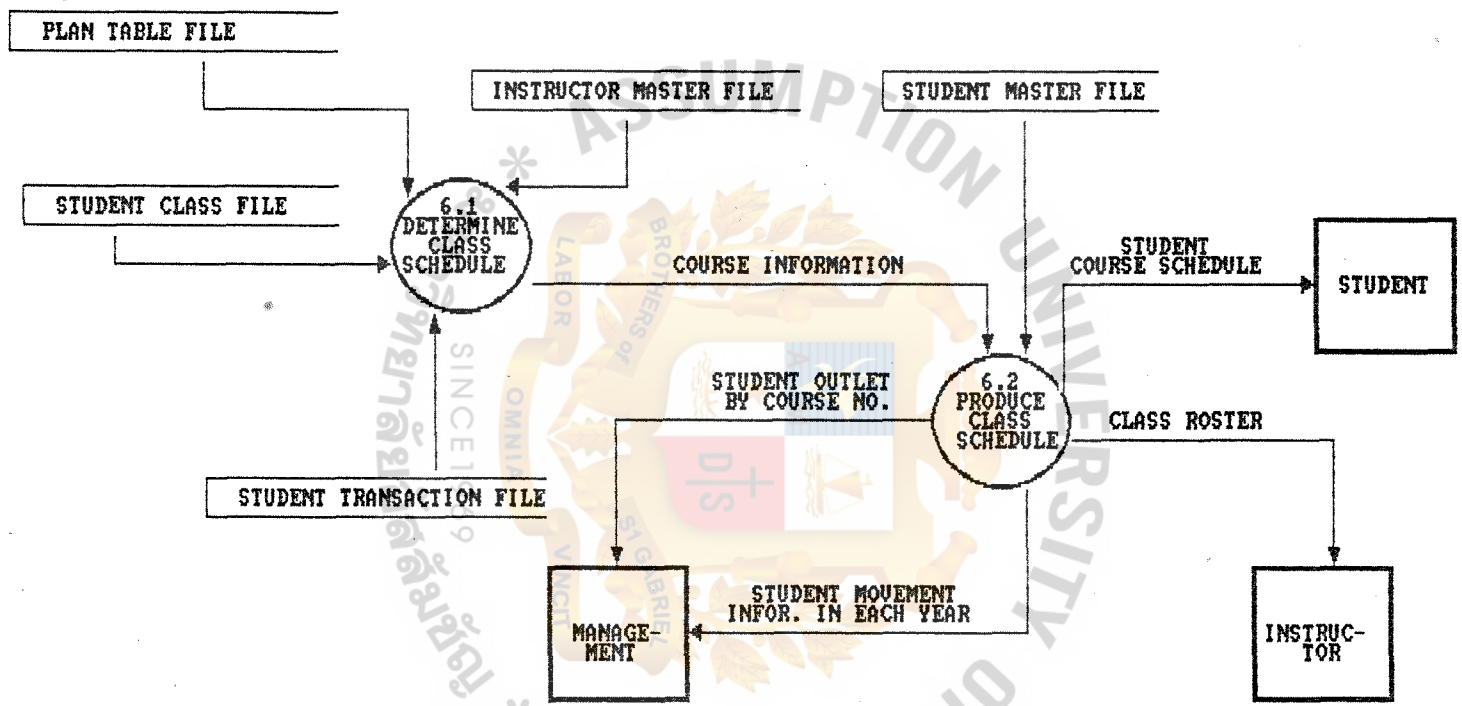


FIGURE B-8 DATA FLOW DIAGRAM LEVEL 1. PROCESS 6 : PREPARE CLASS SCHEDULE

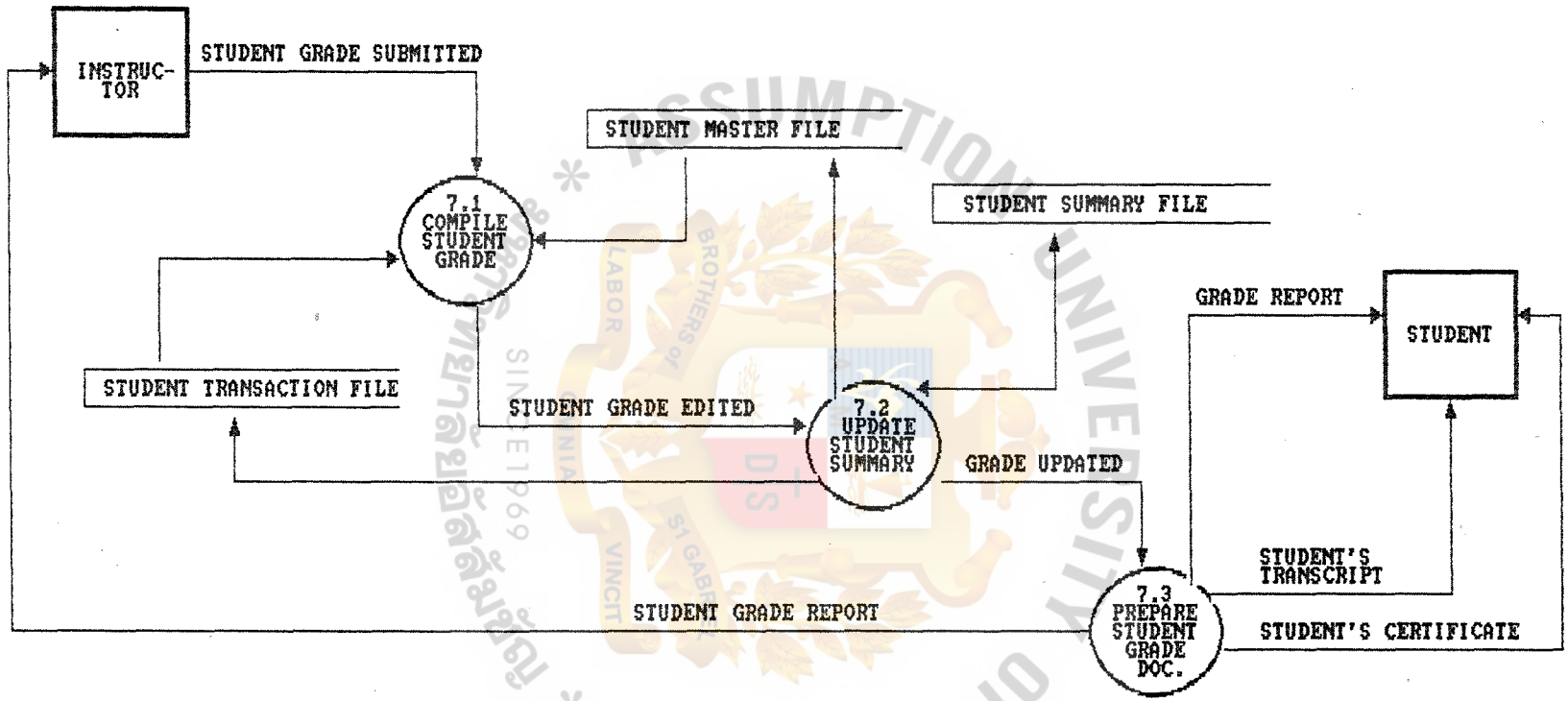


FIGURE B-9 DATA FLOW DIAGRAM LEVEL 1. PROCESS 7 : PROCESS GRADE

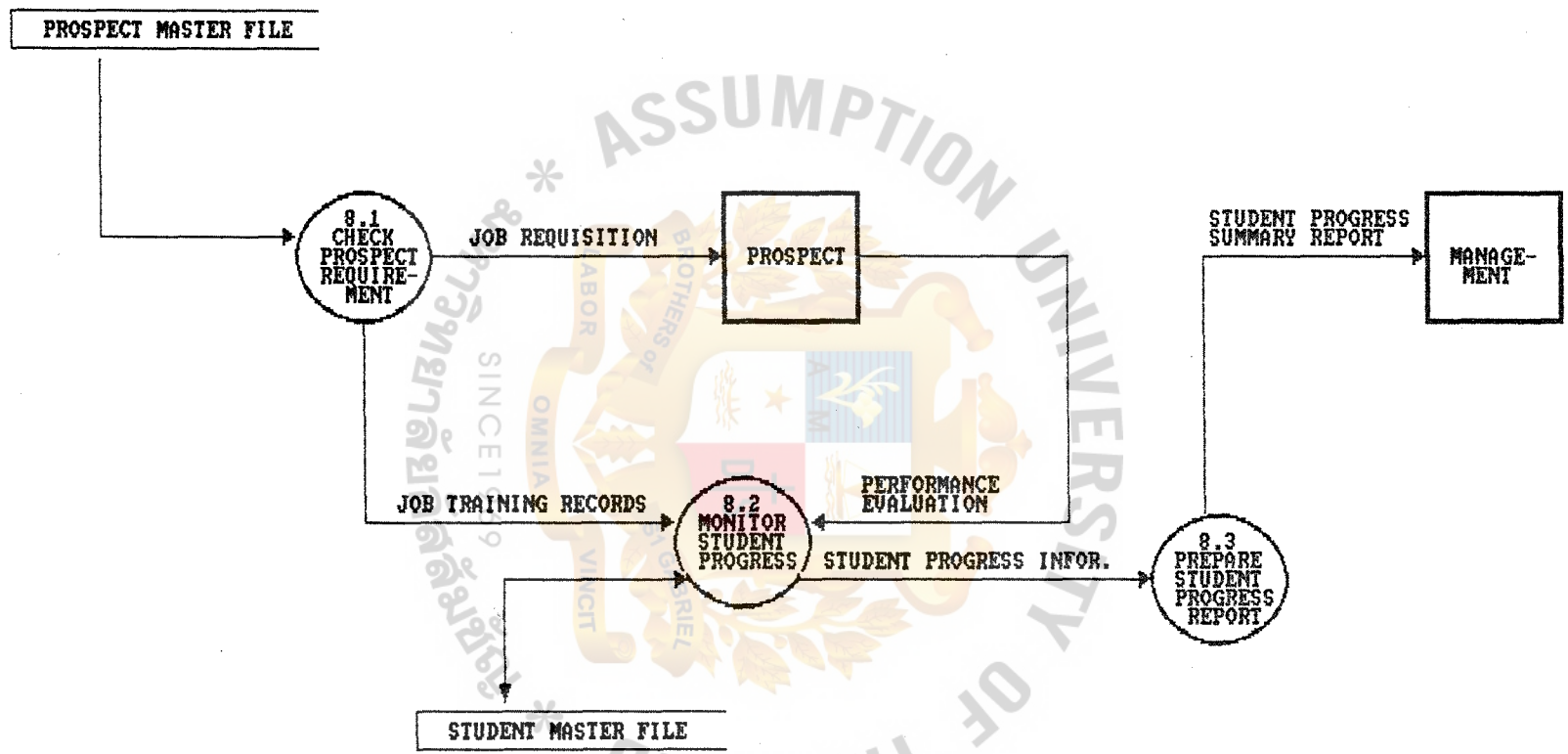


FIGURE B-10 DATA FLOW DIAGRAM LEVEL 1. PROCESS 8 : MONITOR STUDENT PROGRESS

# APPENDIX C
# DATA DICTIONARY

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST. GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

Database INS_MAS Alias Instructor 19:47 10/23/92

Description: Instructor Master File

| Field name | Type | Len | Dec | Key | Options |
|---|---|---|---|---|---|
| INS_NAME | C | 40 | 0 | 0 | |
| INS_NO | N | 3 | 0 | 1 | Picture: 999 |

Figure C-1 Data Dictionary of Data Store: Instructor Master File

Database STU_CLAS Alias Class 19:47 10/23/92

Description: Student Class File

| Field name | Type | Len | Dec | Key | Options |
|---|---|---|---|---|---|
| COURSE_NO | C | 6 | 0 | 1 | |
| COUR_TITLE | C | 30 | 0 | 0 | |
| CREDIT | N | 1 | 0 | 0 | Picture: 9 |
| DAY | C | 12 | 0 | 0 | |
| DURATION | N | 3 | 0 | 0 | Picture: 999 |
| INST_NO | N | 3 | 0 | 0 | Picture: 999 |
| PREREQUIS | C | 6 | 0 | 0 | Picture: 999999 |
| RATE | N | 8 | 2 | 0 | Picture: 99999.99 |
| SECT_NO | N | 1 | 0 | 2 | Picture: 9 |
| TIME | C | 15 | 0 | 0 | Picture: 99.99-99.99 |
| TUITION | N | 7 | 2 | 0 | Picture: 9999.99 |

Figure C-2 Data Dictionary of Data Store: Student Class File

Database STU_MAS  Alias Stu_mas  19:49  10/23/92

Description: Student Master File

| Field name | Type | Len | Dec | Key | Options |
|---|---|---|---|---|---|
| ACC_CREDIT | N | 2 | 0 | 0 | Picture: 99 |
| ACC_GPA | N | 4 | 2 | 0 | Picture: 9.99 |
| AREA_CODE | N | 3 | 0 | 0 | Picture: 999 |
| COM_ADDR1 | C | 40 | 0 | 0 | |
| COM_ADDR2 | C | 40 | 0 | 0 | |
| COM_NAME | C | 40 | 0 | 0 | |
| COM_TEL | C | 15 | 0 | 0 | |
| COM_ZIP | C | 5 | 0 | 0 | |
| CON_ADDR1 | C | 40 | 0 | 0 | |
| CON_ADDR2 | C | 40 | 0 | 0 | |
| CON_NAME | C | 40 | 0 | 0 | |
| CON_TEL | C | 15 | 0 | 0 | |
| CON_ZIP | C | 5 | 0 | 0 | |
| EVA_CODE | C | 1 | 0 | 0 | |
| MEDIA_CODE | N | 1 | 0 | 0 | Picture: 9 |
| MEDIA_NAME | C | 15 | 0 | 0 | |
| OLD_STU_ID | N | 10 | 0 | 0 | Picture: 9999999999 |
| PROS_NO | N | 6 | 0 | 0 | Picture: 999999 |
| RECRUIT | C | 1 | 0 | 0 | |
| STU_ADDR1 | C | 40 | 0 | 0 | |
| STU_ADDR2 | C | 40 | 0 | 0 | |
| STU_BIRTH | D | 8 | 0 | 0 | Picture: @D |
| STU_EVALU | C | 1 | 0 | 0 | Picture: @! |

Figure C-3  Data Dictionary of Data Store: Student Master File

Database STU_MAS  Alias Stu_mas  19:51  10/23/92

Description: Student Master File

| Field name | Type | Len | Dec | Key | Options |
|---|---|---|---|---|---|
| STU_ID | N | 10 | 0 | 1 | Picture: 9999999999 |
| STU_NAME | C | 40 | 0 | 0 | |
| STU_NATION | C | 10 | 0 | 0 | |
| STU_OCCU | C | 15 | 0 | 0 | |
| STU_RELIG | C | 10 | 0 | 0 | |
| STU_SEX | C | 1 | 0 | 0 | |
| STU_STUS | C | 1 | 0 | 0 | |
| STU_TEL | C | 15 | 0 | 0 | |
| STU_TITLE | C | 15 | 0 | 0 | |
| STU_TRAIN | C | 1 | 0 | 0 | Picture: @! |
| STU_ZIP | C | 5 | 0 | 0 | |

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

Figure C-4  Data Dictionary of Data Store: Student Master File (Cont'd)

Database STU_PLAN Alias Plan 19:52 10/23/92

Description: Plan Table File

| Field name | Type | Len | Dec | Key | Options |
|---|---|---|---|---|---|
| CLINIC | N | 8 | 2 | 0 | Picture: 99999.99 |
| LIBRARY | N | 8 | 2 | 0 | Picture: 99999.99 |
| PLAN_NAME | C | 40 | 0 | 0 | |
| PLAN_NO | N | 3 | 0 | 1 | Picture: 999 |
| REGIS | N | 8 | 2 | 0 | Picture: 99999.99 |

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

Figure C-5 Data Dictionary of Data Store: Plan Table File

Database STU_SUM Alias Summary 19:52 10/23/92

Description: Student Summary File

| Field name | Type | Len | Dec | Key | Options |
|---|---|---|---|---|---|
| AMOUNT | N | 8 | 2 | 0 | Picture: 99999.99 |
| AMT1 | N | 8 | 2 | 0 | Picture: 99999.99 |
| AMT2 | N | 8 | 2 | 0 | Picture: 99999.99 |
| AMT3 | N | 8 | 2 | 0 | Picture: 99999.99 |
| AMT4 | N | 8 | 2 | 0 | Picture: 99999.99 |
| BANK_NAME | C | 40 | 0 | 0 | |
| BANK_NO | C | 12 | 0 | 0 | |
| CASH | C | 1 | 0 | 0 | |
| CHEQUE | C | 1 | 0 | 0 | |
| CLINIC | N | 8 | 2 | 0 | Picture: 99999.99 |
| CREDIT | N | 2 | 0 | 0 | Picture: 99 |
| DISCOUNT | N | 5 | 2 | 0 | Picture: 99.99 |
| LIBRARY | N | 8 | 2 | 0 | Picture: 99999.99 |
| NET_AMT | N | 8 | 2 | 0 | Picture: 99999.99 |
| OTHER1 | C | 40 | 0 | 0 | |
| OTHER2 | C | 40 | 0 | 0 | |
| OTHER3 | C | 40 | 0 | 0 | |
| OTHER4 | C | 40 | 0 | 0 | |
| PAY_DATE | D | 8 | 0 | 0 | Picture: @D |
| PER_DIS | N | 8 | 2 | 0 | Picture: 99999.99 |
| REGIS | N | 8 | 2 | 0 | Picture: 99999.99 |
| REG_DATE | D | 8 | 0 | 0 | Picture: @D |
| STU_ID | N | 10 | 0 | 1 | Picture: 9999999999 |
| TERM | C | 4 | 0 | 2 | |
| TUITIONFEE | N | 8 | 2 | 0 | Picture: 99999.99 |

Figure C-6 Data Dictionary of Data Store: Student Summary File

Database STU_TRAN Alias Tran 19:54 10/23/92

Description: Student Transaction File

| Field name | Type | Len | Dec | Key | Options |
|---|---|---|---|---|---|
| COURSE_NO | C | 6 | 0 | 0 | |
| CREDIT | N | 1 | 0 | 0 | Picture: 9 |
| GPA | N | 4 | 2 | 0 | Picture: 9.99 |
| GRADE | C | 2 | 0 | 0 | Picture: @! |
| SECTION_NO | N | 1 | 0 | 0 | Picture: 9 |
| STU_ID | N | 10 | 0 | 1 | Picture: 9999999999 |
| TERM | C | 4 | 0 | 2 | |
| TUITION | N | 8 | 2 | 0 | Picture: 99999.99 |

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

Figure C-7 Data Dictionary of Data Store: Student Transaction File

Database STU_RUN Alias Runno 19:55 10/23/92

Description: Control Running File

| Field name | Type | Len | Dec | Key | Options |
|---|---|---|---|---|---|
| STU_ID | N | 10 | 0 | 1 | Picture: 9999999999 |
| TERM | C | 4 | 0 | 2 | |
| RUN_NO | N | 6 | 0 | 0 | Picture: 999999 |

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

Figure C-8 Data Dictionary of Data Store: Control Running File

GENIFER Data Dictionary

| No | Database | Alias | Description |
|---|---|---|---|
| 1 | INS_MAS | Instructor | Instructor Master File |
| 2 | STU_CLAS | Class | Student Class File |
| 3 | STU_MAS | Stu_mas | Student Master File |
| 4 | STU_PLAN | Plan | Plan Table File |
| 5 | STU_SUM | Summary | Student Summary File |
| 6 | STU_TRAN | Tran | Student Transaction File |
| 7 | STU_RUN | Runno | Control Running File |
| 8 | STU_COST | Cost | Class Cost Control File |
| 9 | CUS_PROS | Prospect | Prospect Master File |

GENIFER File Maintenance Generator

| No | Screen Name | Description | Database |
|---|---|---|---|
| 1 | ST001 | Enter/Update Student Master | STU_MAS |
| 2 | ST002 | Enter/Update Student Class | STU_CLAS |
| 3 | ST003 | Enter/Update Instructor Master | INS_MAS |
| 4 | ST004 | Maintain Class Cost Control | STU_CLAS |
| 5 | ST005 | Enter/Update Plan Table | STU_PLAN |
| 6 | ST006 | Process Student Payment | STU_SUM |
| 7 | ST007 | Process Student Registration | STU_TRAN |
| 8 | ST008 | Process Student Grade | STU_TRAN |
| 9 | ST009 | Post Student History | STU_HIS |
| 10 | ST010 | Enter/Update Prospect Master | CUS_PROS |
| 11 | ST012 | Chop Student Completed Record | STU_CHOP |

Figure C-9 Data Dictionary of System Files

# APPENDIX D
# DATA ACCESS DIAGRAM

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST. GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

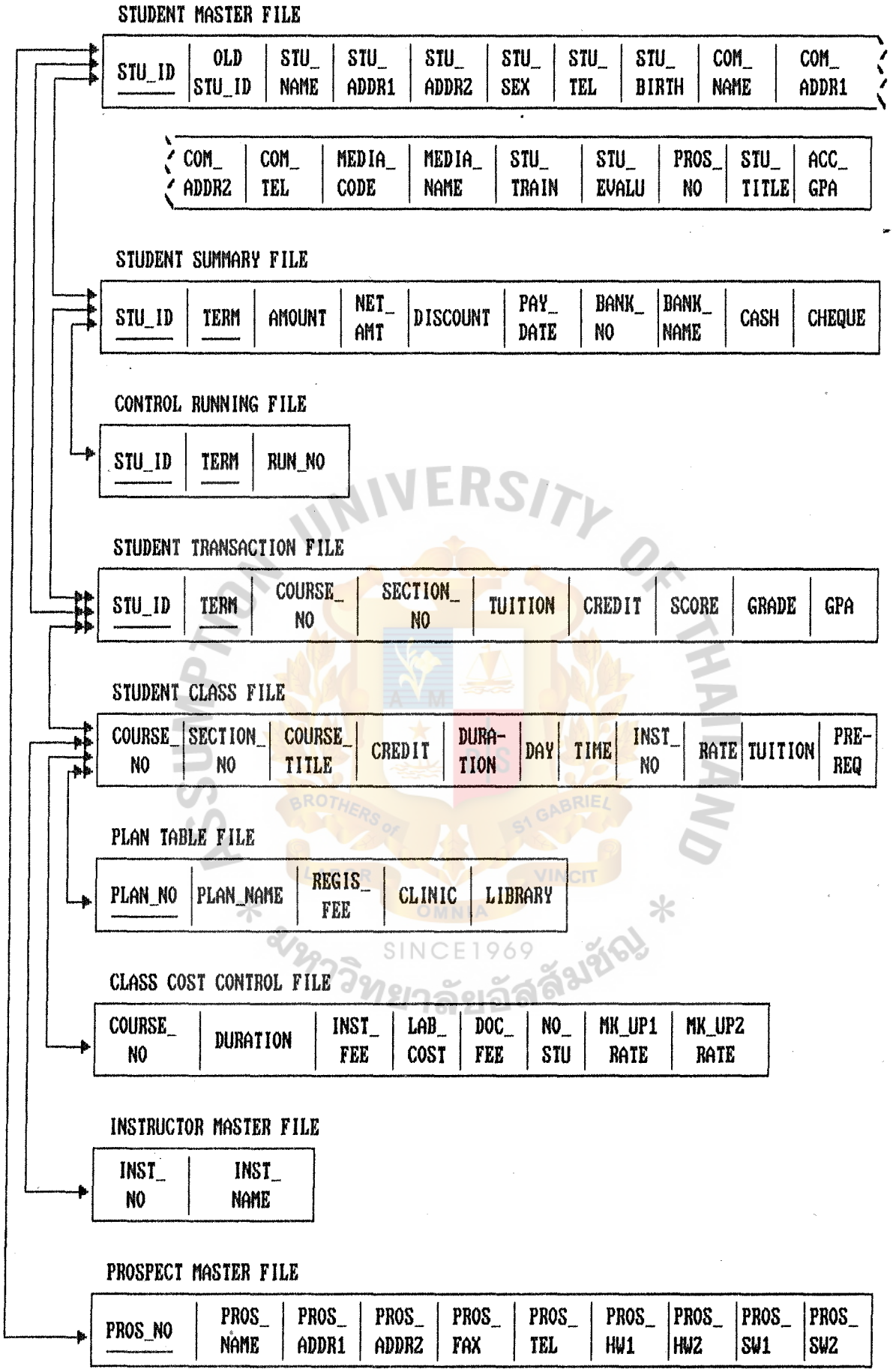


Figure D-1 Data Access Diagram Describing Primary Associations between Files.

# APPENDIX E
# MINISPECIFICATIONS

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST. GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

PROCESS MINISPECIFICATION

Process Name : *Build Class File*

Process Number : *1.0*

Description :

*Determines class planning & class costing information to creating and updating student class file & class cost control file respectively.*

Inputs :

*Class planning from management*
*Class costing information from financial department*
*Cost control analysis report to management*

Outputs :

*Course arrangement to data store : Student class file*
*Class costing information to data store : Class cost control file*

Logic :

*Verify class planning & class costing information. New data, create new records. Old data, update records in files.*

Attachments :

*– None –*

Unresolved Problems :

*– None –*

Figure E-1 Minispecification documentation for process bubble 1.0

PROCESS MINISPECIFICATION

Process Name : *Build Table File*

Process Number : *2.0*

Description :

*Determines plan types information by creating and updating plan table file.*

Inputs :

*Plan type information from management*

Outputs :

*Plan records to data store : Plan table file*

Logic :

*Verify plan type. New plan, create records. Old plan, update records in file.*

Attachments :

*- None -*

Unresolved Problems :

*- None -*

Figure E-2 Minispecification documentation for process bubble 2.0

PROCESS MINISPECIFICATION

Process Name : *Build Master File*

Process Number : *3.0*

Description :

*Maintain student master file, prospect master file, and instructor master file from student application forms, prospect information and instructor list respectively.*

Inputs :

*Student application form from student*
*Prospect information from several medias*
*Instuctor list from management*

Outputs :

*Student information records to data store : Student master file*
*Prospect information records to data store : Prospect master file*
*Instructor code & name to data store : Instructor master file*
*Listing of student information by media type to management*
*Listing of top ranking analysis by student area to management*

Logic :

*Verify student application form to access student registration information. Complete & correct data, release for data entry. Insufficient data, reject. Prospect master file is built by collecting information from several medias (magazine, television, etc.). Instructor list receives from management*

Attachments :

*– None –*

Unresolved Problems :

*Sometime, the office can't get enough information in prospect case. So, any further, update or actual prospect information are incompleted.*

Figure E–3 Minispecification documentation for process bubble 3.0

PROCESS MINISPECIFICATION

Process Name : *Process Registration*

Process Number : *4.0*

Description :

*Student must fill in registration request completely for registering his/her course. If it is valid, student will get registration form.*

Inputs :

*Registration request from student*
*Student ID & name from data store : Student master file*
*Course information from data store : Student class file*
*Plan type information from data store : Plan table file*

Outputs :

*Registration form to student*
*Student registration record to data store : Student transaction file*
*Summary of student regist. record to data store : Student summ. file*
*Student registration in each term to management*
*Tuition fee report fro each course to management*
*Class–offering report to management*

Logic :

*Process student registration record from student registration request. Check registration not more than 7 items, check prerequisite course, confirm student registration record if every steps valid.*

Attachments :

*– None –*

Unresolved Problems :

*Some student wants to take 2 plan types simultaneously. So, he/she need 2 registration request and will get 2 Student IDs.*

Figure E–4 Minispecification documentation for process bubble 4.0

PROCESS MINISPECIFICATION

Process Name : *Process Payment*

Process Number : *5.0*

Description :

*Receive student payment according to amount balance in registration form. Payment type allowed only cash or cheque.*

Inputs :

*Payment from student*
*Student ID & Name from data store : Student master file*
*Student registration records from data store : Student trans. file*
*Summary of student regist. record from data store : Student summ. file*

Outputs :

*Receipt to student*
*New control running no. to data store : Control running file*
*Student payment record to data store : Student summary file*

Logic :

*Check amount payment by verifying student registration records in student transaction file. Payment can be made by cash or cheque.*

Attachments :

*– None –*

Unresolved Problems :

*– None –*

Figure E–5 Minispecification docomentation for process bubble 5.0

PROCESS MINISPECIFICATION

Process Name : *Prepare Class Schedule*

Process Number : *6.0*

Description :

*Determines class schedule for student*

Inputs :

*Student transaction record from data store : Student trans. file*
*Class record from data store : Class file*
*Plan information from data store : Plan table file*
*Instructor name from data store : Instructor master file*

Outputs :

*Course schedule to student*
*Class roster to instructor*

Logic :

*Produce course schedule and class roster by gathering information from plan table file, instructor master file, and student class file*

Attachments :

*– None –*

Unresolved Problems :

*– None –*

Figure E–6 Minispecification documentation for process bubble 6.0

PROCESS MINISPECIFICATION

Process Name : *Process Grade*

Process Number : *7.0*

Description :
*Calculate student grade in each semester.*

Inputs :
*Student trans. record from data store : Student trans. file*
*Student ID & Name for data store : Student master file*
*Student grade from instructor*

Outputs :
*New student grade in student record to data store : Student sum. file*
*Student grade report to instructor*
*Grade report to student*
*Student's transcript to student*
*Student's certificate to student*

Logic :
*Check student grade submitted by instructor. If correct, update student grade to student summary file.*

Attachments :
*–None–*

Unresolved Problems :
*–None–*

Figure E-7 Minispecification documentation for process bubble 7.0

PROCESS MINISPECIFICATION

Process Name : *Monitor student progress*

Process Number : *8.0*

Description :

*Monitor student progress by using performance evaluation to measure result.*

Inputs :

*Prospect name & address to data store : Prospect master file*
*Student ID & name from data store : Student master file*
*Performance evaluation from prospect*

Outputs :

*Job requisition to prospect customer*
*Student progress summary report to management*
*Student progress record to data store : Student master file*

Logic :

*Verify performance evaluation from prospect customer. New student, add student progress record to student master file. Old student, update student progress record to student master file.*

Attachments :

*– None –*

Unresolved Problems :

*– None –*

Figure E-8 Minispecification docomentation for process bubble 8.0

# APPENDIX F
# PROGRAM LISTING
# (PARTIAL)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST. GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

**PROGRAM: ST001.PRG**

```
* Description:  Enter/Update Student Master File
* Author:       Laddawan Chongsiriratanakul with GENIFER
* Date:         10/09/92    Time: 18:32
* Portions Copyright (c) 1986 Bytel Corporation. Use licensed by Bytel Corp.

* MAINTENANCE OPTIONS:
  * Duplic. keys: no
  * Modify key:   no
  * Auto add:     yes
  * Partial key:  yes
  * Comments:     yes
  * Help screens: yes

PROCEDURE ST001

* FILES
  * Student Master File
  database1 = 'STU_MAS'
  index_fil1 = 'STU_MAS'

* VARIABLES
  * Key in database STU_MAS
    stu_id = 0
  * Fields in database STU_MAS
    old_stu_id = 0
    stu_name = ''
    stu_addr1 = ''
    stu_addr2 = ''
    stu_zip = ''
    stu_tel = ''
    stu_stus = ''
    stu_sex = ''
    stu_birth = date()
    stu_nation = ''
    stu_relig = ''
    com_name = ''
    com_addr1 = ''
    com_addr2 = ''
    com_zip = ''
    com_tel = ''
    stu_occu = ''
    stu_title = ''
    media_code = 0
    media_name = ''
    con_addr1 = ''
    con_addr2 = ''
    con_zip = ''
    con_tel = ''
    stu_train = ''
    stu_evalu = ''
    recruited = ''
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
  * Flags
    abort_rec = .F.
    dupl_rec = .F.
    empty = .F.
    files_ok = .T.
    del_rec = .F.
    valid_rec = .T.
    show_all = .T.
    filter_on1 = .F.
    set_filter = .F.
  * Others
    key1 = 'str(m->stu_id,10,0)'
    index1 = 'str(stu_id,10,0)'
    null_key1 = '         0'
    comp_key1 = 'trim(str(m->stu_id,10,0))'

    list1 = 'stu_id,old_stu_id,stu_name'
    pak1 = .F.
    choice = ''
    option = ''
    filt_str = ''
    filter1 = ''
    scr_num = 1
    num = '1'
    max_screen = 1
    record_no = 0


* ENVIRONMENT
  set exact off
  set heading off
  set bell off
  set deleted on

* SCREEN
  do disp_scr

* OPEN FILES
  do chk_fils
  if .not. files_ok
    return
  endif
  index_str = index_fil1
  select 1
  use &database1 index &index_str
  select stu_mas
  record_no = recno()
  * database is empty if eof() = .T.
  empty = eof()
  if empty
    do disp_msg with 'Database empty'
  endif

* PROCESSING LOOP
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
do while scr_num >= 1
  record_no = recno()
  * preserve option if in add mode
  if empty .or. (scr_num = max_screen .and. option = 'A')
    option = 'A'
  else
    do load_var
    do disp_rec
    option = 'M'
  endif
  do get_optn with 'Ret/Beg/End/Next/Prev/Skip/Modify/Add/Copy/' + ;
                   'Del/List/Filt/Tally/Help/Quit','RBENPSMACDLFTHQ',option

  do case
  * Add
    case option = 'A'
      do add
  * Beginning
    case option = 'B'
      go top
  * Copy
    case option = 'C'
      do add
  * Delete
    case option = 'D'
      do delete
  * End
    case option = 'E'
      go bottom
  * Filter
    case option = 'F'
      do filter
  * Help
    case option = 'H'
      do help
      show_all = .T.
      do disp_scr
  * List
    case option = 'L'
      do list with list&num
      show_all = .T.
      do disp_scr
      go record_no
  * Modify
    case option = 'M'
      do modify
  * Next
    case option = 'N'
      skip
      if eof()
        go bottom
        do disp_msg with 'Last record'
      endif
  * Previous
    case option = 'P'
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
        skip -1
        if bof()
          go top
          do disp_msg with 'First record'
        endif
    * Retrieve
      case option = 'R'
        do retrieve
    * Skip
      case option = 'S'
        do skip
    * Tally
      case option = 'T'
        do tally
    endcase

  * Quit
    if option = 'Q'
      scr_num = scr_num - 1
      num = str(scr_num,1)
      if scr_num <> 0
        select &num
        empty = .F.
        show_all = .T.
        do disp_scr
      endif
    endif
  enddo (while scr_num >= 1)

  if del_rec
    do pack_all
  endif
  close databases
return

* PROCEDURES (listed alphabetically)

procedure add
* Add a new record (if option = 'A') or copy the current
* record (if option = 'C')
  if option = 'A'
    do init_key
    do init_fld
  endif
  * The following "get record" loop can be exited either when a valid
  * record is entered (valid_rec = .T.), or when entry is aborted by
  * (1) blank key, (2) duplicate key (if not allowed), or (3) abort
  * request from the validation procedure (if called).
  abort_rec = .F.
  valid_rec = .F.
  do while .not. (valid_rec .or. abort_rec)
    do get_key
    do get_flds
    read
    comp_key = comp_key&num
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
      if null_key&num = &comp_key
        * blank key
        abort_rec = .T.
      else
        do chk_dupl
        if dupl_rec
          abort_rec = .T.
        else
          valid_rec = .T.
        endif (dupl_rec)
      endif (null_key = comp_key)
    enddo (while .not. (valid_rec .or. abort_rec))
    if valid_rec
      do save_rec with .T.
    else
      * break out of "Add" if record invalid
      if empty
        option = 'Q'
      else
        option = 'M'
        if .not. dupl_rec
          go record_no
        endif
      endif (empty)
    endif (valid_rec)
  return

  procedure chk_dupl
  * Set dupl_rec to .F. if key is a duplicate, or to .T. otherwise
    key = key&num
    if .not. empty
      seek &key
    endif
    if option = 'M'
      dupl_rec = record_no <> recno() .and. .not. eof()
    else
      dupl_rec = .not. eof()
    endif
    if dupl_rec
      do disp_msg with chr(7) + 'Duplicate key not allowed'
    else
      if .not. empty
        go record_no
      endif
    endif dupl_rec
  return

  procedure chk_fils
  * Set files_ok to .F. and display a message if a file is missing;
  * create an index file if one does not exist.
    close databases
    file_num = 1
    do while file_num <= max_screen .and. files_ok
      seq = str(file_num,1)
      if .not. file (database&seq + '.dbf')
```

```
      do disp_msg with 'File ' + database&seq + ' not found'
      files_ok = .F.
    endif
    if files_ok .and. .not. file (index_fil&seq + '.ntx')
      @ 23,00
      @ 23,00 say 'Indexing'
      db = database&seq
      ind = index_fil&seq
      use &db
      ind_key = index&seq
      index on &ind_key to &ind
      use
    endif
    file_num = file_num + 1
  enddo (while file_num <= max_screen .and. files_ok)
  @ 23,00
  @ 24,00
return

procedure clr_flds
* Clear screen field areas
  @  3,48 say space (10)
  @  4,18 say space (40)
  @  5,18 say space (40)
  @  6,18 say space (40)
  @  6,73 say space (5)
  @  7,18 say space (15)
  @  8,18 say space (1)
  @  8,58 say space (1)
  @  9,18 say space (8)
  @  9,43 say space (10)
  @  9,68 say space (10)
  @ 11,18 say space (40)
  @ 12,18 say space (40)
  @ 13,18 say space (40)
  @ 13,73 say space (5)
  @ 14,18 say space (15)
  @ 15,18 say space (15)
  @ 15,50 say space (15)
  @ 16,18 say space (1)
  @ 16,64 say space (15)
  @ 18,18 say space (40)
  @ 19,18 say space (40)
  @ 19,73 say space (5)
  @ 20,18 say space (15)
  @ 21,18 say space (1)
  @ 21,45 say space (1)
  @ 21,72 say space (1)
return

procedure delete
* Delete current record upon user approval
  choice = 'N'
  do get_optn with 'Delete (Y/N)','YN',choice
  if choice = 'Y'
```

```
    delete
    * reposition to the next record
    skip
    * if last record deleted, go to beginning of database
    if eof()
      go top
    endif
    if eof() .and. filter_on&num
      * if no records left in filter, remove the filter
      filt_str = filter&num
      set filter to &filt_str
      filter_on&num = .F.
      @ 0,0 say space(9)
      go top
    endif (eof() .and. filter_on&num)
    if eof()
      * quit if last record deleted and database becomes empty
      option = 'Q'
    endif
    del_rec = .T.
    pak&num = .T.
  endif (choice = 'Y')
return

procedure disp_msg
parameters message
* Display "message" + '...' at line 23; wait for entry of any key
  @ 23,00
  @ 22,79
  wait message + '...'
return

procedure disp_rec
* Load fields from database and display record
  do get_key
  do get_flds
  clear gets
  show_all = .F.
return

procedure disp_scr
* Display stationary part of screen
  clear
  set color to n/r
  @  1,16 say 'I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)'
  set color to w
  @  1,72 say date()
  set color to w+
  @  2, 0 to  22,79 double
  @  3, 2 say 'Student ID.'
  @  3,32 say 'Old Student ID.'
  @  4, 2 say 'Name'
  @  5, 2 say 'Address'
  @  6,62 say 'Post Code'
  @  7, 2 say 'Telephone'
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
@  8, 2 say 'Marital Status'
set color to w
@  8,20 say '<'
set color to w+
@  8,21 say 'S'
set color to w
@  8,22 say '-Single,'
set color to w+
@  8,31 say 'M'
set color to w
@  8,32 say '-Married,'
set color to w+
@  8,42 say 'O'
set color to w
@  8,43 say '-Other>'
set color to w+
@  8,54 say 'Sex'
set color to w
@  8,60 say '<'
set color to w+
@  8,61 say 'M'
set color to w
@  8,62 say '-Male,'
set color to w+
@  8,69 say 'F'
set color to w
@  8,70 say '-Female>'
set color to w+
@  9, 2 say 'Date Of Birth                    Nationality'
@  9,58 say 'Religion'
@ 11, 2 say 'Office Name'
@ 12, 2 say 'Office Address'
@ 13,62 say 'Post Code'
@ 14, 2 say 'Telephone'
@ 15, 2 say 'Occupation                                  T'
@ 15,42 say 'itle'
@ 16, 2 say 'Media Code'
set color to w
@ 16,20 say '<'
set color to w+
@ 16,21 say '1'
set color to w
@ 16,22 say '-Magazine,'
set color to w+
@ 16,33 say '2'
set color to w
@ 16,34 say '-Television,'
set color to w+
@ 16,46 say '3'
set color to w
@ 16,47 say '-Radio,'
set color to w+
@ 16,55 say '4'
set color to w
@ 16,56 say '-Other>'
```

ASSUMPTION UNIVERSITY OF THAILAND
A M D S
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
  set color to w+
  @ 18, 2 say 'Contact Address'
  @ 19,62 say 'Post Code'
  @ 20, 2 say 'Telephone'
  @ 21, 2 say 'Training Job        <Y,N>  Evaluation Gra'
  @ 21,42 say 'de   <A,B,C,D>       Recruited   <Y,N>'
  set color to w
  if filter_on&num
    @ 0,0 say 'FILTER ON'
  else
    @ 0,0 say space(9)
  endif
return

procedure filter
* Set filter on database
  choice = 'Y'
  do get_optn with 'Set Filter (Yes/No/Cancel)','YNC',choice
  if choice = 'Y'
    set_filter = .T.
    do init_key
    do init_fld
    do get_key
    do get_flds
    read
    filt_str = ''
    if m->stu_id <> 0
      stu_id = str(m->stu_id,10,0)
      filt_str = filt_str + 'stu_id = &stu_id .and.'
    endif
    if '' <> trim(m->stu_name)
      filt_str = filt_str + 'stu_name = trim("&stu_name") .and.'
    endif
    if '' <> trim(m->stu_addr1)
      filt_str = filt_str + 'stu_addr1 = trim("&stu_addr1") .and.'
    endif
    if '' <> trim(m->stu_addr2)
      filt_str = filt_str + 'stu_addr2 = trim("&stu_addr2") .and.'
    endif
    if '' <> trim(m->stu_zip)
      filt_str = filt_str + 'stu_zip = trim("&stu_zip") .and.'
    endif
    if '' <> trim(m->stu_tel)
      filt_str = filt_str + 'stu_tel = trim("&stu_tel") .and.'
    endif
    if '' <> trim(m->stu_stus)
      filt_str = filt_str + 'stu_stus = trim("&stu_stus") .and.'
    endif
    if '' <> trim(m->stu_sex)
      filt_str = filt_str + 'stu_sex = trim("&stu_sex") .and.'
    endif
    if .not. (m->stu_birth <> m->stu_birth .or. .not. (m->stu_birth = m->stu_birth))
      stu_birth = dtoc(m->stu_birth)
      filt_str = filt_str + 'stu_birth = ctod("&stu_birth") .and.'
    endif
```

```
if '' <> trim(m->stu_nation)
  filt_str = filt_str + 'stu_nation = trim("&stu_nation") .and.'
endif
if '' <> trim(m->stu_relig)
  filt_str = filt_str + 'stu_relig = trim("&stu_relig") .and.'
endif
if '' <> trim(m->com_name)
  filt_str = filt_str + 'com_name = trim("&com_name") .and.'
endif
if '' <> trim(m->com_addr1)
  filt_str = filt_str + 'com_addr1 = trim("&com_addr1") .and.'
endif
if '' <> trim(m->com_addr2)
  filt_str = filt_str + 'com_addr2 = trim("&com_addr2") .and.'
endif
if '' <> trim(m->com_zip)
  filt_str = filt_str + 'com_zip = trim("&com_zip") .and.'
endif
if '' <> trim(m->com_tel)
  filt_str = filt_str + 'com_tel = trim("&com_tel") .and.'
endif
if '' <> trim(m->stu_occu)
  filt_str = filt_str + 'stu_occu = trim("&stu_occu") .and.'
endif
if '' <> trim(m->stu_title)
  filt_str = filt_str + 'stu_title = trim("&stu_title") .and.'
endif
if m->media_code <> 0
  media_code = str(m->media_code,1,0)
  filt_str = filt_str + 'media_code = &media_code .and.'
endif
if '' <> trim(m->media_name)
  filt_str = filt_str + 'media_name = trim("&media_name") .and.'
endif
if '' <> trim(m->con_addr1)
  filt_str = filt_str + 'con_addr1 = trim("&con_addr1") .and.'
endif
if '' <> trim(m->con_addr2)
  filt_str = filt_str + 'con_addr2 = trim("&con_addr2") .and.'
endif
if '' <> trim(m->con_zip)
  filt_str = filt_str + 'con_zip = trim("&con_zip") .and.'
endif
if '' <> trim(m->con_tel)
  filt_str = filt_str + 'con_tel = trim("&con_tel") .and.'
endif
if '' <> trim(m->stu_train)
  filt_str = filt_str + 'stu_train = trim("&stu_train") .and.'
endif
if '' <> trim(m->stu_evalu)
  filt_str = filt_str + 'stu_evalu = trim("&stu_evalu") .and.'
endif
if '' <> trim(m->recruited)
  filt_str = filt_str + 'recruited = trim("&recruited") .and.'
endif
```

```
    if '' = trim(filt_str)
      filt_str = filter&num
      set filter to &filt_str
      filter_on&num = .F.
    else
      if '' <> trim(filter&num)
        filt_str = filter&num + '.and.' + filt_str
      endif
      filt_str = substr(filt_str,1,len(filt_str)-6)
      set filter to &filt_str
      go top
      filter_on&num = .not. eof()
      if .not. filter_on&num
        do disp_msg with 'No records match the filter'
        filt_str = filter&num
        set filter to &filt_str
        go record_no
      endif
    endif ('' = trim(filt_str))
  endif (choice = 'Y')
  if choice = 'C'
    filt_str = filter&num
    set filter to &filt_str
    filter_on&num = .F.
  endif (choice = 'C')
  if filter_on&num
    @ 0,0 say 'FILTER ON'
  else
    @ 0,0 say space(9)
  endif
  set_filter = .F.
return

procedure get_flds
* Get field variables
  @  3,48 get m->old_stu_id
  @  4,18 get m->stu_name
  @  5,18 get m->stu_addr1
  @  6,18 get m->stu_addr2
  @  6,73 get m->stu_zip
  @  7,18 get m->stu_tel
  @  8,18 get m->stu_stus
  @  8,58 get m->stu_sex
  @  9,18 get m->stu_birth picture '@D'
  @  9,43 get m->stu_nation
  @  9,68 get m->stu_relig
  @ 11,18 get m->com_name
  @ 12,18 get m->com_addr1
  @ 13,18 get m->com_addr2
  @ 13,73 get m->com_zip
  @ 14,18 get m->com_tel
  @ 15,18 get m->stu_occu
  @ 15,50 get m->stu_title
  @ 16,18 get m->media_code picture '9'
  @ 16,64 get m->media_name
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
  @ 18,18 get m->con_addr1
  @ 19,18 get m->con_addr2
  @ 19,73 get m->con_zip
  @ 20,18 get m->con_tel
  @ 21,18 get m->stu_train picture '@!'
  @ 21,45 get m->stu_evalu picture '@!'
  @ 21,72 get m->recruited picture '@!'
return

procedure get_key
* Get key variables
  @  3,18 get m->stu_id picture '9999999999'
return

procedure get_optn
parameters message, choices, choice
* Display the string "message" on line 23; get a character
* (defaulted to "choice"), validate it against "choices" and
* return in "choice" (must be a memory variable, not literal).
* If choice = 'A' display choice without accepting it.
  @ 23,00 say message + '? '
  @ 23,len(message)+3
  char = ' '
  do while .not. char $ choices
    char = choice
    @ 23,len(message) + 2 get char picture '!'
    if choice <> 'A'
      read
    else
      clear gets
    endif
  enddo (while .not. char $ choices)
  choice = char
return

procedure help
* Display help screens
  clear
  @ 03,00
  text
```

| Option | Database maintenance action |
|---|---|
| Add | add a record to the database |
| Beg | go to beginning of database |
| Copy | duplicate current record |
| Del | delete current record |
| End | go to end of database |
| Filt | set filter on database |
| List | display records on screen |
| Modify | edit current record |
| Next | go to next record |
| Prev | go to previous record |
| Quit | terminate current activity |
| Ret | retrieve a record by key |
| Skip | move up or down by a specified number of records |

```
                Tally    count records
  endtext
  do disp_msg with 'More'
  clear
  @ 05,00
  text
                  White key     Screen editing action
                  ---------     ---------------------
                  -->           character right
                  <--           character left
                  up arrow      previous field
                  down arrow    next field
                  PgDn, PgUp    accept screen
                  End           next word/field
                  Home          previous word/field
                  Del           delete character
                  Ins           insert on/off toggle
  endtext
  do disp_msg with 'OK'
  clear
return

procedure init_fld
* Clear field variables
  old_stu_id = 0
  stu_name = space(40)
  stu_addr1 = space(40)
  stu_addr2 = space(40)
  stu_zip = space(5)
  stu_tel = space(15)
  stu_stus = space(1)
  stu_sex = space(1)
  stu_birth = ctod('  /  /  ')
  stu_nation = space(10)
  stu_relig = space(10)
  com_name = space(40)
  com_addr1 = space(40)
  com_addr2 = space(40)
  com_zip = space(5)
  com_tel = space(15)
  stu_occu = space(15)
  stu_title = space(15)
  media_code = 0
  media_name = space(15)
  con_addr1 = space(40)
  con_addr2 = space(40)
  con_zip = space(5)
  con_tel = space(15)
  stu_train = space(1)
  stu_evalu = space(1)
  recruited = space(1)
return

procedure init_key
* Clear key variables
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
  stu_id = 0
return

procedure list
parameters list_items
* List records beginning from current record
  do while .T.
    clear
    list off next 20 stu_id,old_stu_id,stu_name
    if eof()
      do disp_msg with 'OK'
      exit
    endif
    choice = 'Y'
    do get_optn with 'More','YN',choice
    if choice = 'N'
      exit
    endif
  enddo (while .T.)
return

procedure load_var
* Copy fields from database record to memory variables
  stu_id = stu_id
  old_stu_id = old_stu_id
  stu_name = stu_name
  stu_addr1 = stu_addr1
  stu_addr2 = stu_addr2
  stu_zip = stu_zip
  stu_tel = stu_tel
  stu_stus = stu_stus
  stu_sex = stu_sex
  stu_birth = stu_birth
  stu_nation = stu_nation
  stu_relig = stu_relig
  com_name = com_name
  com_addr1 = com_addr1
  com_addr2 = com_addr2
  com_zip = com_zip
  com_tel = com_tel
  stu_occu = stu_occu
  stu_title = stu_title
  media_code = media_code
  media_name = media_name
  con_addr1 = con_addr1
  con_addr2 = con_addr2
  con_zip = con_zip
  con_tel = con_tel
  stu_train = stu_train
  stu_evalu = stu_evalu
  recruited = recruited
return

procedure modify
* Modify current record
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
* The following "get record" loop is exited when either (1) a valid record
* is entered (valid_rec = .T.), or (2) entry is aborted by a duplicate
* key (when not allowed), or a request from the validation procedure.
  valid_rec = .F.
  abort_rec = .F.
  do while .not. (valid_rec .or. abort_rec)
    do get_key
    do get_flds
    read
    comp_key = comp_key&num
    if null_key&num = &comp_key
      abort_rec = .T.
    else
      do chk_dupl
      if dupl_rec
        abort_rec = .T.
      else
        valid_rec = .T.
      endif (dupl_rec)
    endif (null_key = comp_key)
  enddo (while .not. (valid_rec .or. abort_rec))
  if valid_rec
    do save_rec with .F.
  else
    go record_no
  endif (valid_rec)
return

procedure pack_all
* Pack deleted records with user confirmation
  choice = 'N'
  do get_optn with 'Pack all deleted records (Y/N)','YN',choice
  if choice = 'Y'
    @ 23,00
    @ 23,00 say 'Packing, please wait'
    do while scr_num < max_screen
      scr_num = scr_num + 1
      num = str(scr_num,1)
      if pak&num
        select &num
        pack
      endif
    enddo (while scr_num < max_screen)
  endif
return

procedure repl_rec
* Replace database fields with memory variables
  replace stu_id with m->stu_id
  replace old_stu_id with m->old_stu_id
  replace stu_name with m->stu_name
  replace stu_addr1 with m->stu_addr1
  replace stu_addr2 with m->stu_addr2
  replace stu_zip with m->stu_zip
  replace stu_tel with m->stu_tel
```

```
  replace stu_stus with m->stu_stus
  replace stu_sex with m->stu_sex
  replace stu_birth with m->stu_birth
  replace stu_nation with m->stu_nation
  replace stu_relig with m->stu_relig
  replace com_name with m->com_name
  replace com_addr1 with m->com_addr1
  replace com_addr2 with m->com_addr2
  replace com_zip with m->com_zip
  replace com_tel with m->com_tel
  replace stu_occu with m->stu_occu
  replace stu_title with m->stu_title
  replace media_code with m->media_code
  replace media_name with m->media_name
  replace con_addr1 with m->con_addr1
  replace con_addr2 with m->con_addr2
  replace con_zip with m->con_zip
  replace con_tel with m->con_tel
  replace stu_train with m->stu_train
  replace stu_evalu with m->stu_evalu
  replace recruited with m->recruited
return

procedure retrieve
* Accept key and seek record; if not found reposition to record_no
  do clr_flds
  do init_key
  do get_key
  read
  comp_key = comp_key&num
  if null_key&num = &comp_key
    * blank key
    return
  endif
  key = key&num
  seek trim(&key)
  if eof()
    do disp_msg with 'Not found'
    go record_no
  endif
return

procedure save_rec
parameters new_rec
* If new_rec: append record currently in memory to database;
* if .not. new_rec: replace database record record_no with memory fields.
  choice = 'Y'
  do get_optn with 'Save (Y/N)','YN',choice
  if choice = 'Y'
    if new_rec
      append blank
    endif
    do repl_rec
    empty = .F.
    if filter_on&num
```

```
      * Check to see if record matches filter
      record_no = recno()
      skip
      skip -1
      if recno() <> record_no
        go top
        if eof()
          * Remove filter
          filt_str = filter&num
          set filter to &filt_str
          filter_on&num = .F.
          @ 0,10 say space(9)
          go record_no
        endif
      endif
    endif (filter_on&num)
  else
    if .not. empty
      go record_no
    endif
  endif (choice = 'Y')
return

procedure skip
* Move forward/backward several records
  skip_no = 0
  @ 23,70 say ' Recs' get skip_no picture '@Z 999'
  read
  skip skip_no
  if eof()
    go bottom
    do load_var
    do disp_rec
    do disp_msg with 'Last Record'
  endif
  if bof()
    go top
    do load_var
    do disp_rec
    do disp_msg with 'First record'
  endif
return

procedure tally
* Count and display number of records in database
  @ 23,00
  @ 23,00 say 'Counting, please wait'
  count to number_
  do disp_msg with 'Count: ' + str (m->number_,6) + ' records'
  go record_no
return

* EOF ST001.PRG
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

**PROGRAM: ST004.PRG**

```
* Description:  Cost Control Analysis For Each Course
* Author:       Laddawan Chongsiriratanakul with GENIFER
* Date:         10/13/92    Time: 19:40
* Portions Copyright (c) 1986 Bytel Corporation. Use licensed by Bytel Corp.

* INQUIRY OPTIONS:
  * Partial key:  yes
  * Comments:     yes
  * Help screens: yes

PROCEDURE ST004

* FILES
  * Student Class File
  database1 = 'STU_CLAS'
  index_fil1 = 'STU_CLAS'

* VARIABLES
  * Key in database STU_CLAS
    course_no = ''
    sect_no = 0
  * Fields in database STU_CLAS
    duration = 0
    rete = 0

  * Flags
    empty = .F.
    files_ok = .T.
    show_all = .T.
    filter_on1 = .F.
    set_filter = .F.
  * Others
    key1 = 'm->course_no + str(m->sect_no,1,0)'
    index1 = 'course_no + str(sect_no,1,0)'
    null_key1 = '       0'
    comp_key1 = 'trim(m->course_no + str(m->sect_no,1,0))'

    list1 = 'course_no,duration'
    choice = ''
    option = ''
    filt_str = ''
    filter1 = ''
    scr_num = 1
    num = '1'
    max_screen = 1
    record_no = 0
    mcourse_no = ''
    mduration = 0
    qty_record = 0
    rete = 0
```

```
    rate1 = 0
    rate2 = 0
    rate3 = 0
    rate4 = 0
    rate5 = 0
    inst1 = 0
    inst2 = 0
    inst3 = 0
    inst4 = 0
    inst5 = 0
    lab_cost = 0
    stu_men1 = 20
    document = 0
    doc_cost = 0
    stu_min = 0
    mtuition = 0
    mark_up1 = 40
    mark_up2 = 40
    percent1 = 0
    percent2 = 0
    tot_cost1 = 0
    tot_cost2 = 0
    tot_cost3 = 0
    total_rate  = 0

* ENVIRONMENT
  set exact off
  set heading off
  set bell off
  set deleted on

* SCREEN
  do disp_scr

* OPEN FILES
  do chk_fils
  if .not. files_ok
    return
  endif
  index_str = index_fil1
  select 1
  use &database1 index &index_str alias class
  select class
  record_no = recno()
  * database is empty if eof() = .T.
  empty = eof()
  if empty
    do disp_msg with 'Database empty'
    option = 'A'
  endif

* PROCESSING LOOP
  do while scr_num >= 1
    record_no = recno()
    if option <> 'M'
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
   do load_var
   do disp_rec
endif
option = 'R'
do get_optn with 'Ret/Beg/End/Next/Prev/Skip/Modify/' + ;
                 'List/Filt/Tally/Help/Quit','RBENPSMLFTHQ',option

do case
* Beginning
  case option = 'B'
    go top
* End
  case option = 'E'
    go bottom
* Filter
  case option = 'F'
    do filter
* Help
  case option = 'H'
    do help
    show_all = .T.
    do disp_scr
* List
  case option = 'L'
    do list with list&num
    show_all = .T.
    do disp_scr
    go record_no
* Modify
  case option = 'M'
    do modify
* Next
  case option = 'N'
    skip qty_record
    if eof()
      go bottom
      do disp_msg with 'Last record'
    endif
* Previous
  case option = 'P'
    skip -qty_record
    if bof()
      go top
      do disp_msg with 'First record'
    endif
* Retrieve
  case option = 'R'
    do retrieve
* Skip
  case option = 'S'
    do skip
* Tally
  case option = 'T'
    do tally
endcase
```

```
  * Quit
    if option = 'Q'
      scr_num = scr_num - 1
      num = str(scr_num,1)
      if scr_num <> 0
        select &num
        empty = .F.
        show_all = .T.
        do disp_scr
      endif
    endif

  enddo (while scr_num >= 1)
  close databases
return

* PROCEDURES (listed alphabetically)

procedure chk_fils
* Set files_ok to .F. and display a message if a file is missing;
* create an index file if one does not exist.
  close databases
  file_num = 1
  do while file_num <= max_screen .and. files_ok
    seq = str(file_num,1)
    if .not. file (database&seq + '.dbf')
      do disp_msg with 'File ' + database&seq + ' not found'
      files_ok = .F.
    endif
    if files_ok .and. .not. file (index_fil&seq + '.ntx')
      @ 23,00 say 'Indexing'
      db = database&seq
      ind = index_fil&seq
      use &db
      ind_key = index&seq
      index on &ind_key to &ind
      use
    endif
    file_num = file_num + 1
  enddo (file_num <= max_screen .and. files_ok)
  @ 23,00
  @ 24,00
return

procedure clr_cal
* Clear computed screen areas
  @  6,40 say space(9)
  @  7,40 say space(9)
  @  8,40 say space(9)
  @  9,40 say space(9)
  @ 10,40 say space(9)
  @ 10,60 say space(9)
  @ 12,60 say space(9)
  @ 13,60 say space(9)
```

```
  @ 14,60 say space(9)
  @ 15,60 say space(9)
  @ 16,60 say space(9)
  @ 18,60 say space(9)
  @ 19,60 say space(9)
  @ 20,67 say space(2)
return

procedure clr_flds
* Clear screen field areas
  @  5,66 say space(3)
  @  6,30 say space(9)
  @  7,30 say space(9)
  @  8,30 say space(9)
  @  9,30 say space(9)
  @ 10,30 say space(9)
  @ 11,60 say space(9)
  @ 12,28 say space(2)
  @ 12,46 say space(9)
  @ 14,28 say space(2)
  @ 18,43 say space(2)
return

procedure disp_cal
* Evaluate and display computed fields
  inst1 = rate1 * mduration
  @  6,40 say inst1 picture '99,999.99'
  inst2 = rate2 * mduration
  @  7,40 say inst2 picture '99,999.99'
  inst3 = rate3 * mduration
  @  8,40 say inst3 picture '99,999.99'
  inst4 = rate4 * mduration
  @  9,40 say inst4 picture '99,999.99'
  inst5 = rate5 * mduration
  @ 10,40 say inst5 picture '99,999.99'
  total_rate = (inst1+inst2+inst3+inst4+inst5)/qty_record
  @ 10,60 say total_rate picture '99,999.99'
  doc_cost = stu_men1 * document
  set color to u
  @ 12,60 say doc_cost picture '99,999.99'
  set color to w
  tot_cost1 = total_rate + lab_cost + doc_cost
  @ 13,60 say tot_cost1 picture '99,999.99'
  percent1 = (tot_cost1 * mark_up1) / 100
  set color to u
  @ 14,60 say percent1 picture '99,999.99'
  set color to w
  tot_cost2 = tot_cost1 + percent1
  @ 15,60 say tot_cost2 picture '99,999.99'
  mtuition = tot_cost2 / stu_men1
  @ 16,60 say mtuition picture '99,999.99'
  percent2 = (tot_cost1 * mark_up2) / 100
  set color to u
  @ 18,60 say percent2 picture '99,999.99'
  tot_cost3 = tot_cost1 + percent2
```

```
  @ 19,60 say tot_cost3 picture '99,999.99'
  stu_min = tot_cost3 / mtuition
  @ 20,67 say stu_min picture '99'
  set color to w
return

procedure disp_msg
parameters message
* Display "message" + '...' at line 23; wait for entry of any key
  @ 23,00
  @ 22,79
  wait message + '...'
return

procedure disp_rec
* Load fields from database and display record
  do load_var
  do disp_cal
  do get_key
  do get_flds
  clear gets

  show_all = .F.
return

procedure disp_scr
* Display stationary part of screen
  clear
  set color to n/r
  @  1,16 say 'I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)'
  set color to w
  @  1,72 say date()
  set color to w+
  @  3, 5 to  21,74
  @  2, 0 to  22,79 double
  @  4,11 say 'Course No.'
  @  5,11 say 'Duration'
  @  5,70 say 'hr.'
  @  6,11 say 'Instructor Fees 1:'
  @  7,27 say '2:'
  @  8,27 say '3:'
  @  9,27 say '4:'
  @ 10,27 say '5:'
  @ 10,70 say 'Baht'
  @ 11,11 say 'Lab Cost'
  @ 11,70 say 'Baht'
  @ 12,11 say 'Student In Class      Document Fee'
  @ 12,70 say 'Baht'
  @ 13,11 say 'Total Cost Control'
  @ 13,70 say 'Baht'
  @ 14,11 say 'Standard Mark-Up    %'
  @ 14,70 say 'Baht'
  @ 15,11 say 'Total Course Fee'
  @ 15,70 say 'Baht'
  @ 16,11 say 'Tuition Per Person'
```

```
  @ 16,70 say 'Baht'
  @ 18,11 say 'Course Fee (Available) Mark-Up     %'
  @ 18,70 say 'Baht'
  @ 19,11 say 'Total Course Fee (Available)'
  @ 19,70 say 'Baht'
  @ 20,11 say 'Minimum Student (Available)'
  @ 20,70 say 'Mens'
  set color to w
  if filter_on&num
    @ 0,10 say 'FILTER ON'
  else
    @ 0,10 say space(9)
  endif
return

procedure filter
* Set filter on database
  choice = 'Y'
  do get_optn with 'Set Filter (Yes/No/Cancel)','YNC',choice
  if choice = 'Y'
    set_filter = .T.
    do init_key
    do init_fld
    do clr_cal
    do get_key
    do get_flds
    read
    filt_str = ''
    if '' <> trim(m->course_no)
      filt_str = filt_str + 'course_no = trim("&course_no") .and.'
    endif
    if m->duration <> 0
      duration = str(m->duration,3,0)
      filt_str = filt_str + 'duration = &duration .and.'
    endif
    if '' = trim(filt_str)
      filt_str = filter&num
      set filter to &filt_str
      filter_on&num = .F.
    else
      if '' <> trim(filter&num)
        filt_str = filter&num + '.and.' + filt_str
      endif
      filt_str = substr(filt_str,1,len(filt_str)-6)
      set filter to &filt_str
      go top
      filter_on&num = .not. eof()
      if .not. filter_on&num
        do disp_msg with 'No records match the filter'
        filt_str = filter&num
        set filter to &filt_str
        go record_no
      endif
    endif ('' = trim(filt_str))
  endif (choice = 'Y')
```

```
  if choice = 'C'
    filt_str = filter&num
    set filter to &filt_str
    filter_on&num = .F.
  endif (choice = 'C')
  if filter_on&num
    @ 0,10 say 'FILTER ON'
  else
    @ 0,10 say space(9)
  endif
  set_filter = .F.
return

procedure get_flds
* Get field variables
  @  5,66 say m->duration picture '999'
  @  6,30 get rate1 picture '99,999.99'
  @  7,30 get rate2 picture '99,999.99'
  @  8,30 get rate3 picture '99,999.99'
  @  9,30 get rate4 picture '99,999.99'
  @ 10,30 get rate5 picture '99,999.99'
  @ 11,60 get lab_cost picture '99,999.99'
  @ 12,28 get stu_men1 picture '99'
  @ 12,46 get document picture '99,999.99'
  @ 14,28 get mark_up1 picture '99'
  @ 18,43 get mark_up2 picture '99'
return

procedure get_key
* Get key variables
  @  4,22 get m->course_no
return

procedure get_optn
parameters message, choices, choice
* Display the string "message" on line 23; get a character
* (defaulted to "choice"), validate it against "choices" and
* return in "choice" (must be a memory variable, not literal).
* If choice = 'A' display choice without accepting it.
  @ 23,00 say message + '? '
  @ 23,len(message)+3
  char = ' '
  do while .not. char $ choices
    char = choice
    @ 23,len(message) + 2 get char picture '!'
    if choice <> 'A'
      read
    else
      clear gets
    endif
  enddo (while .not. char $ choices)
  choice = char
return

procedure help
```

```
* Display help screens
  clear
  @ 03,00
  text
                  Option    Database inquiry action
                  ------    -----------------------
                  Beg       go to beginning of database
                  End       go to end of database
                  Filt      set filter on database
                  List      display records on screen
                  Next      go to next record
                  Prev      go to previous record
                  Quit      terminate current activity
                  Ret       retrieve a record by key
                  Skip      move up or down by a specified number of records
                  Tally     count records
  endtext
  do disp_msg with 'More'
  clear
  @ 05,00
  text
                    White key     Screen editing action
                    ---------     ---------------------
                    -->           character right
                    <--           character left
                    up arrow      previous field
                    down arrow    next field
                    PgDn, PgUp    accept screen
                    End           next word/field
                    Home          previous word/field
                    Del           delete character
                    Ins           insert on/off toggle
  endtext
  do disp_msg with 'OK'
  clear
return

procedure init_fld
* Clear field variables
    duration  =  0
    rate1     =  0
    rate2     =  0
    rate3     =  0
    rate4     =  0
    rate5     =  0
    inst1     =  0
    inst2     =  0
    inst3     =  0
    inst4     =  0
    inst5     =  0
    lab_cost  =  0
    stu_men1  = 20
    document  =  0
    doc_cost  =  0
    stu_min   =  0
```

```
    mtuition  =  0
    mark_up1  = 40
    mark_up2  = 40
    percent1  =  0
    percent2  =  0
    tot_cost1 =  0
    tot_cost2 =  0
    tot_cost3 =  0
    total_rate=  0

return

procedure init_key
* Clear key variables
  course_no = space(6)
return

procedure list
parameters list_items
* List records beginning from current record
  do while .T.
    clear
    list off next 20 course_no,duration
    if eof()
      do disp_msg with 'OK'
      exit
    endif
    choice = 'Y'
    do get_optn with 'More','YN',choice
    if choice = 'N'
      exit
    endif
  enddo (while .T.)
return

procedure load_var
* Copy fields from database record to memory variables
  course_no = course_no
  duration  = duration
  mcourse_no = course_no
  mduration  = duration
  qty_record = 0
  total_rate = 0
  runno = str(qty_record,1)
  do while course_no = mcourse_no .and. qty_record <= 5 .and. .not. eof()
     qty_record = qty_record + 1
     runno = str(qty_record,1)
     rate&runno = rate
*    total_rate = total_rate + rate
     skip
  enddo
  skip -qty_record
return

procedure retrieve
```

```
* Accept key and seek record; if not found reposition to record_no
  do clr_flds
  do clr_cal
  do init_key
  do get_key
  read
  comp_key = comp_key&num
  if null_key&num = &comp_key
    * blank key
    return
  endif
  key = key&num
  seek trim(&key)
  if eof()
    do disp_msg with 'Not found'
    go record_no
  endif
return

procedure skip
* Move forward/backward several records
  skip_no = 0
  @ 23,70 say '  Recs' get skip_no picture '@Z 999'
  read
  skip skip_no
  if eof()
    go bottom
    do load_var
    do disp_rec
    do disp_msg with 'Last Record'
  endif
  if bof()
    go top
    do load_var
    do disp_rec
    do disp_msg with 'First record'
  endif
return

procedure modify
* Modify current record
* The following "get record" loop is exited when either (1) a valid record
* is entered (valid_rec = .T.), or (2) entry is aborted by a duplicate
* key (when not allowed), or a request from the validation procedure.
  valid_rec = .F.
  abort_rec = .F.
*  do while .not. (valid_rec .or. abort_rec)
    do get_flds
    read
    do disp_cal
    valid_rec = .T.
*  enddo (while .not. (valid_rec .or. abort_rec))
return

procedure tally
```

ASSUMPTION UNIVERSITY OF THAILAND
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
* Count and display number of records in database
  @ 23,00 say 'Counting, please wait' + space(59)
  count to number_
  do disp_msg with 'Count: ' + str (m->number_,6) + ' records'
  go record_no
return

* EOF ST004.PRG
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

**PROGRAM: ST006.PRG**

```
* Description:  Process Student Payment As Request
* Author:       Laddawan Chongsiriratanakul with GENIFER
* Date:         10/22/92    Time: 16:42
* Portions Copyright (c) 1986 Bytel Corporation. Use licensed by Bytel Corp.

* MAINTENANCE OPTIONS:
  * Duplic. keys: no
  * Modify key:   no
  * Auto add:     no
  * Partial key:  yes
  * Comments:     yes
  * Help screens: yes

PROCEDURE ST006

* FILES
  * Summary File For Period
  database1 = 'STU_SUM'
  index_fil1 = 'STU_SUM'

* VARIABLES
  * Key in database STU_SUM
    stu_id = 0
    term = ''
  * Fields in database STU_SUM
    pay_date = date()
    regis = 0
    tuitionfee = 0
    library = 0
    clinic = 0
    other1 = ''
    amt1 = 0
    other2 = ''
    amt2 = 0
    other3 = ''
    amt3 = 0
    other4 = ''
    amt4 = 0
    net_amt = 0
    cash = ''
    cheque = ''
    bank_name = ''
    bank_no = ''
    name = space(35)

  * Flags
    abort_rec = .F.
    dupl_rec = .F.
    empty = .F.
    files_ok = .T.
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
    del_rec = .F.
    valid_rec = .T.
    show_all = .T.
    filter_on1 = .F.
    set_filter = .F.
  * Others
    key1 = 'str(m->stu_id,10,0) + m->term'
    index1 = 'str(stu_id,10,0) + term'
    null_key1 = '          0'
    comp_key1 = 'trim(str(m->stu_id,10,0) + m->term)'

    list1 = 'stu_id,term,pay_date,regis,tuitionfee,library,clinic'
    pak1 = .F.
    choice = ''
    option = ''
    filt_str = ''
    filter1 = ''
    scr_num = 1
    num = '1'
    max_screen = 1
    record_no = 0


* ENVIRONMENT
  set exact off
  set heading off
  set bell off
  set deleted on

* SCREEN
  do disp_scr

* OPEN FILES
  do chk_fils
  if .not. files_ok
    return
  endif
  index_str = index_fil1
  select 1
  use &database1 index &index_str alias summary
  select 2
  use stu_mas index stu_mas
  select summary
  record_no = recno()
  * database is empty if eof() = .T.
  empty = eof()
  if empty
    do disp_msg with 'Database empty'
  endif

* PROCESSING LOOP
  do while scr_num >= 1
    record_no = recno()
    if empty
      option = 'A'
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
else
  do load_var
  do disp_rec
  option = 'M'
endif
do get_optn with 'Ret/Beg/End/Next/Prev/Skip/Modify/Add/Copy/' + ;
                 'Del/List/Filt/Tally/Help/Quit','RBENPSMACDLFTHQ',option

do case
* Add
  case option = 'A'
    do add
* Beginning
  case option = 'B'
    go top
* Copy
  case option = 'C'
    do add
* Delete
  case option = 'D'
    do delete
* End
  case option = 'E'
    go bottom
* Filter
  case option = 'F'
    do filter
* Help
  case option = 'H'
    do help
    show_all = .T.
    do disp_scr
* List
  case option = 'L'
    do list with list&num
    show_all = .T.
    do disp_scr
    go record_no
* Modify
  case option = 'M'
    do modify
* Next
  case option = 'N'
    skip
    if eof()
      go bottom
      do disp_msg with 'Last record'
    endif
* Previous
  case option = 'P'
    skip -1
    if bof()
      go top
      do disp_msg with 'First record'
    endif
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
    * Retrieve
      case option = 'R'
        do retrieve
    * Skip
      case option = 'S'
        do skip
    * Tally
      case option = 'T'
        do tally
    endcase

  * Quit
    if option = 'Q'
      scr_num = scr_num - 1
      num = str(scr_num,1)
      if scr_num <> 0
        select &num
        empty = .F.
        show_all = .T.
        do disp_scr
      endif
    endif
  enddo (while scr_num >= 1)

  if del_rec
    do pack_all
  endif
  close databases
return

* PROCEDURES (listed alphabetically)

procedure add
* Add a new record (if option = 'A') or copy the current
* record (if option = 'C')
  if option = 'A'
    do init_key
    do init_fld
  endif
  * The following "get record" loop can be exited either when a valid
  * record is entered (valid_rec = .T.), or when entry is aborted by
  * (1) blank key, (2) duplicate key (if not allowed), or (3) abort
  * request from the validation procedure (if called).
  abort_rec = .F.
  valid_rec = .F.
  do while .not. (valid_rec .or. abort_rec)
    do get_key
    do get_flds
    read
    comp_key = comp_key&num
    if null_key&num = &comp_key
      * blank key
      abort_rec = .T.
    else
      do chk_dupl
```

```
      if dupl_rec
        abort_rec = .T.
      else
        valid_rec = .T.
      endif (dupl_rec)
    endif (null_key = comp_key)
  enddo (while .not. (valid_rec .or. abort_rec))
  if valid_rec
    do save_rec with .T.
  else
    * break out of "Add" if record invalid
    if empty
      option = 'Q'
    else
      option = 'M'
      if .not. dupl_rec
        go record_no
      endif
    endif (empty)
  endif (valid_rec)
return

procedure chk_dupl
* Set dupl_rec to .F. if key is a duplicate, or to .T. otherwise
  key = key&num
  if .not. empty
    seek &key
  endif
  if option = 'M'
    dupl_rec = record_no <> recno() .and. .not. eof()
  else
    dupl_rec = .not. eof()
  endif
  if dupl_rec
    do disp_msg with chr(7) + 'Duplicate key not allowed'
  else
    if .not. empty
      go record_no
    endif
  endif dupl_rec
return

procedure chk_fils
* Set files_ok to .F. and display a message if a file is missing;
* create an index file if one does not exist.
  close databases
  file_num = 1
  do while file_num <= max_screen .and. files_ok
    seq = str(file_num,1)
    if .not. file (database&seq + '.dbf')
      do disp_msg with 'File ' + database&seq + ' not found'
      files_ok = .F.
    endif
    if files_ok .and. .not. file (index_fil&seq + '.ntx')
      @ 23,00
```

```
      @ 23,00 say 'Indexing'
      db = database&seq
      ind = index_fil&seq
      use &db
      ind_key = index&seq
      index on &ind_key to &ind
      use
    endif
    file_num = file_num + 1
  enddo (while file_num <= max_screen .and. files_ok)
  @ 23,00
  @ 24,00
return

procedure clr_flds
* Clear screen field areas
  @  3,40 say space (35)
  @  4,16 say space (8)
  @  9,66 say space (8)
  @ 10,66 say space (8)
  @ 11,66 say space (8)
  @ 12,66 say space (8)
  @ 13,16 say space (40)
  @ 13,66 say space (8)
  @ 14,16 say space (40)
  @ 14,66 say space (8)
  @ 15,16 say space (40)
  @ 15,66 say space (8)
  @ 16,16 say space (40)
  @ 16,66 say space (8)
  @ 18,66 say space (8)
  @ 20, 3 say space (1)
  @ 21, 3 say space (1)
  @ 21,19 say space (40)
  @ 21,65 say space (12)
return

procedure delete
* Delete current record upon user approval
  choice = 'N'
  do get_optn with 'Delete (Y/N)','YN',choice
  if choice = 'Y'
    delete
    * reposition to the next record
    skip
    * if last record deleted, go to beginning of database
    if eof()
      go top
    endif
    if eof() .and. filter_on&num
      * if no records left in filter, remove the filter
      filt_str = filter&num
      set filter to &filt_str
      filter_on&num = .F.
      @ 0,0 say space(9)
```

```
      go top
    endif (eof() .and. filter_on&num)
    if eof()
      * quit if last record deleted and database becomes empty
      option = 'Q'
    endif
    del_rec = .T.
    pak&num = .T.
  endif (choice = 'Y')
return

procedure disp_msg
parameters message
* Display "message" + '...' at line 23; wait for entry of any key
  @ 23,00
  @ 22,79
  wait message + '...'
return

procedure disp_rec
* Load fields from database and display record
  do get_key
  do get_flds
  clear gets
  show_all = .F.
return

procedure disp_scr
* Display stationary part of screen
  clear
  @  1,13 say 'I M A C (INFORMATION MANAGEMENT ACADEMIC'
  @  1,54 say 'CENTRE)        10/22/92'
  @  2, 0 say '╔═══════════════════════════════════════════════'
  @  2,40 say '═══════════════════════════════════════╗'
  @  3, 0 say '│ Student ID. :                    Term :'
  @  3,78 say '│'
  @  4, 0 say '│ Payment Date:'
  @  4,78 say '│'
  @  5, 0 say '│'
  @  5,78 say '│'
  @  6, 0 say '│ ┌──────────────────────────────────────'
  @  6,40 say '──────────────────────┬──────────────────┐│'
  @  7, 0 say '│ │                    Description'
  @  7,60 say '│    Amount        ││'
  @  8, 0 say '│ │──────────────────────────────────────'
  @  8,40 say '──────────────────────┼──────────────────┤│'
  @  9, 0 say '│ │ Registration Fee'
  @  9,60 say '│                  ││'
  @ 10, 0 say '│ │ Tuition Fee'
  @ 10,60 say '│                  ││'
  @ 11, 0 say '│ │ Library Fee'
  @ 11,60 say '│                  ││'
  @ 12, 0 say '│ │ Physical Examination Or Clinic Fee'
  @ 12,60 say '│                  ││'
  @ 13, 0 say '│ │ Other    1)'
```

```
  @ 13,60 say '│                   │║'
  @ 14, 0 say '║ │          2)'
  @ 14,60 say '│                   │║'
  @ 15, 0 say '║ │          3)'
  @ 15,60 say '│                   │║'
  @ 16, 0 say '║ │          4)'
  @ 16,60 say '│                   │║'
  @ 17, 0 say '║ │'
  @ 17,60 say '│                   │║'
  @ 18, 0 say '║ │ The sum of ░░░░░░░░░░░░░░░░░░░░░░░░░'
  @ 18,40 say '░░░░░░░░░░░░░░░░░░░░│                │║'
  @ 19, 0 say '║ └──────────────────────────────────────'
  @ 19,40 say '────────────────── └────────────────┘ ║'
  @ 20, 0 say '║ [ ] Cash'
  @ 20,78 say '║'
  @ 21, 0 say '║ [ ] Cheque   Bank'
  @ 21,61 say 'No.        ║'
  @ 22, 0 say '╚═══════════════════════════════════════'
  @ 22,40 say '═══════════════════════════════════════╝'
  if filter_on&num
    @ 0,0 say 'FILTER ON'
  else
    @ 0,0 say space(9)
  endif
return

procedure filter
* Set filter on database
  choice = 'Y'
  do get_optn with 'Set Filter (Yes/No/Cancel)','YNC',choice
  if choice = 'Y'
    set_filter = .T.
    do init_key
    do init_fld
    do get_key
    do get_flds
    read
    filt_str = ''
    if m->stu_id <> 0
      stu_id = str(m->stu_id,10,0)
      filt_str = filt_str + 'stu_id = &stu_id .and.'
    endif
    if '' <> trim(m->term)
      filt_str = filt_str + 'term = trim("&term") .and.'
    endif
    if .not. (m->pay_date <> m->pay_date .or. .not. (m->pay_date = m->pay_date))
      pay_date = dtoc(m->pay_date)
      filt_str = filt_str + 'pay_date = ctod("&pay_date") .and.'
    endif
    if m->regis <> 0
      regis = str(m->regis,8,2)
      filt_str = filt_str + 'regis = &regis .and.'
    endif
    if m->tuitionfee <> 0
      tuitionfee = str(m->tuitionfee,8,2)
```

```
  filt_str = filt_str + 'tuitionfee = &tuitionfee .and.'
endif
if m->library <> 0
  library = str(m->library,8,2)
  filt_str = filt_str + 'library = &library .and.'
endif
if m->clinic <> 0
  clinic = str(m->clinic,8,2)
  filt_str = filt_str + 'clinic = &clinic .and.'
endif
if '' <> trim(m->other1)
  filt_str = filt_str + 'other1 = trim("&other1") .and.'
endif
if m->amt1 <> 0
  amt1 = str(m->amt1,8,2)
  filt_str = filt_str + 'amt1 = &amt1 .and.'
endif
if '' <> trim(m->other2)
  filt_str = filt_str + 'other2 = trim("&other2") .and.'
endif
if m->amt2 <> 0
  amt2 = str(m->amt2,8,2)
  filt_str = filt_str + 'amt2 = &amt2 .and.'
endif
if '' <> trim(m->other3)
  filt_str = filt_str + 'other3 = trim("&other3") .and.'
endif
if m->amt3 <> 0
  amt3 = str(m->amt3,8,2)
  filt_str = filt_str + 'amt3 = &amt3 .and.'
endif
if '' <> trim(m->other4)
  filt_str = filt_str + 'other4 = trim("&other4") .and.'
endif
if m->amt4 <> 0
  amt4 = str(m->amt4,8,2)
  filt_str = filt_str + 'amt4 = &amt4 .and.'
endif
if m->net_amt <> 0
  net_amt = str(m->net_amt,8,2)
  filt_str = filt_str + 'net_amt = &net_amt .and.'
endif
if '' <> trim(m->cash)
  filt_str = filt_str + 'cash = trim("&cash") .and.'
endif
if '' <> trim(m->cheque)
  filt_str = filt_str + 'cheque = trim("&cheque") .and.'
endif
if '' <> trim(m->bank_name)
  filt_str = filt_str + 'bank_name = trim("&bank_name") .and.'
endif
if '' <> trim(m->bank_no)
  filt_str = filt_str + 'bank_no = trim("&bank_no") .and.'
endif
if '' = trim(filt_str)
```

```
      filt_str = filter&num
      set filter to &filt_str
      filter_on&num = .F.
    else
      if '' <> trim(filter&num)
        filt_str = filter&num + '.and.' + filt_str
      endif
      filt_str = substr(filt_str,1,len(filt_str)-6)
      set filter to &filt_str
      go top
      filter_on&num = .not. eof()
      if .not. filter_on&num
        do disp_msg with 'No records match the filter'
        filt_str = filter&num
        set filter to &filt_str
        go record_no
      endif
    endif ('' = trim(filt_str))
  endif (choice = 'Y')
  if choice = 'C'
    filt_str = filter&num
    set filter to &filt_str
    filter_on&num = .F.
  endif (choice = 'C')
  if filter_on&num
    @ 0,0 say 'FILTER ON'
  else
    @ 0,0 say space(9)
  endif
  set_filter = .F.
return

procedure get_flds
* Get field variables
  select 2
  seek str(m->stu_id,10,0)
  if found()
     name = substr(stu_name,1,35)
  endif
  select 1
  @  3,40 say name
  @  4,16 get m->pay_date picture '@D'
  @  9,66 say m->regis picture '999,999.99'
  @ 10,66 say m->tuitionfee picture '999,999.99'
  @ 11,66 say m->library picture '999,999.99'
  @ 12,66 say m->clinic picture '999,999.99'
  @ 13,16 say m->other1
  @ 13,66 say m->amt1 picture '999,999.99'
  @ 14,16 say m->other2
  @ 14,66 say m->amt2 picture '999,999.99'
  @ 15,16 say m->other3
  @ 15,66 say m->amt3 picture '999,999.99'
  @ 16,16 say m->other4
  @ 16,66 say m->amt4 picture '999,999.99'
  @ 18,66 say m->net_amt picture '999,999.99'
```

```
  @ 20, 3 get m->cash picture '@!'
  @ 21, 3 get m->cheque picture '@!'
  @ 21,19 get m->bank_name
  @ 21,65 get m->bank_no
return

procedure get_key
* Get key variables
  @  3,16 get m->stu_id picture '9999999999'
  @  3,35 get m->term
return

procedure get_optn
parameters message, choices, choice
* Display the string "message" on line 23; get a character
* (defaulted to "choice"), validate it against "choices" and
* return in "choice" (must be a memory variable, not literal).
* If choice = 'A' display choice without accepting it.
  @ 23,00 say message + '? '
  @ 23,len(message)+3
  char = ' '
  do while .not. char $ choices
    char = choice
    @ 23,len(message) + 2 get char picture '!'
    if choice <> 'A'
      read
    else
      clear gets
    endif
  enddo (while .not. char $ choices)
  choice = char
return

procedure help
* Display help screens
  clear
  @ 03,00
  text
                Option    Database maintenance action
                ------    ---------------------------
                Add       add a record to the database
                Beg       go to beginning of database
                Copy      duplicate current record
                Del       delete current record
                End       go to end of database
                Filt      set filter on database
                List      display records on screen
                Modify    edit current record
                Next      go to next record
                Prev      go to previous record
                Quit      terminate current activity
                Ret       retrieve a record by key
                Skip      move up or down by a specified number of records
                Tally     count records
  endtext
```

```
  do disp_msg with 'More'
  clear
  @ 05,00
  text
                    White key     Screen editing action
                    ---------     ----------------------
                    -->           character right
                    <--           character left
                    up arrow      previous field
                    down arrow    next field
                    PgDn, PgUp    accept screen
                    End           next word/field
                    Home          previous word/field
                    Del           delete character
                    Ins           insert on/off toggle
  endtext
  do disp_msg with 'OK'
  clear
return

procedure init_fld
* Clear field variables
  pay_date = ctod('  /  /  ')
  regis = 0.00
  tuitionfee = 0.00
  library = 0.00
  clinic = 0.00
  other1 = space(40)
  amt1 = 0.00
  other2 = space(40)
  amt2 = 0.00
  other3 = space(40)
  amt3 = 0.00
  other4 = space(40)
  amt4 = 0.00
  net_amt = 0.00
  cash = space(1)
  cheque = space(1)
  bank_name = space(40)
  bank_no = space(12)
return

procedure init_key
* Clear key variables
  stu_id = 0
  term = space(4)
return

procedure list
parameters list_items
* List records beginning from current record
  do while .T.
    clear
    list off next 20 stu_id,term,pay_date,regis,tuitionfee,library,clinic
    if eof()
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
      do disp_msg with 'OK'
      exit
    endif
    choice = 'Y'
    do get_optn with 'More','YN',choice
    if choice = 'N'
      exit
    endif
  enddo (while .T.)
return

procedure load_var
* Copy fields from database record to memory variables
  stu_id = stu_id
  term = term
  pay_date = pay_date
  regis = regis
  tuitionfee = tuitionfee
  library = library
  clinic = clinic
  other1 = other1
  amt1 = amt1
  other2 = other2
  amt2 = amt2
  other3 = other3
  amt3 = amt3
  other4 = other4
  amt4 = amt4
  net_amt = net_amt
  cash = cash
  cheque = cheque
  bank_name = bank_name
  bank_no = bank_no

return

procedure modify
* Modify current record
* The following "get record" loop is exited when either (1) a valid record
* is entered (valid_rec = .T.), or (2) entry is aborted by a duplicate
* key (when not allowed), or a request from the validation procedure.
  valid_rec = .F.
  abort_rec = .F.
  do while .not. (valid_rec .or. abort_rec)
    do get_flds
    read
    valid_rec = .T.
  enddo (while .not. (valid_rec .or. abort_rec))
  if valid_rec
    do save_rec with .F.
  else
    go record_no
  endif (valid_rec)
return
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
procedure pack_all
* Pack deleted records with user confirmation
  choice = 'N'
  do get_optn with 'Pack all deleted records (Y/N)','YN',choice
  if choice = 'Y'
    @ 23,00
    @ 23,00 say 'Packing, please wait'
    do while scr_num < max_screen
      scr_num = scr_num + 1
      num = str(scr_num,1)
      if pak&num
        select &num
        pack
      endif
    enddo (while scr_num < max_screen)
  endif
return

procedure repl_rec
* Replace database fields with memory variables
  replace stu_id with m->stu_id
  replace term with m->term
  replace pay_date with m->pay_date
  replace regis with m->regis
  replace tuitionfee with m->tuitionfee
  replace library with m->library
  replace clinic with m->clinic
  replace other1 with m->other1
  replace amt1 with m->amt1
  replace other2 with m->other2
  replace amt2 with m->amt2
  replace other3 with m->other3
  replace amt3 with m->amt3
  replace other4 with m->other4
  replace amt4 with m->amt4
  replace net_amt with m->net_amt
  replace cash with m->cash
  replace cheque with m->cheque
  replace bank_name with m->bank_name
  replace bank_no with m->bank_no
return

procedure retrieve
* Accept key and seek record; if not found reposition to record_no
  do clr_flds
  do init_key
  do get_key
  read
  comp_key = comp_key&num
  if null_key&num = &comp_key
    * blank key
    return
  endif
  key = key&num
  seek trim(&key)
```

```
  if eof()
    do disp_msg with 'Not found'
    go record_no
  endif
return

procedure save_rec
parameters new_rec
* If new_rec: append record currently in memory to database;
* if .not. new_rec: replace database record record_no with memory fields.
  choice = 'Y'
  do get_optn with 'Save (Y/N)','YN',choice
  if choice = 'Y'
    if new_rec
      append blank
    endif
    do repl_rec
    empty = .F.
    if filter_on&num
      * Check to see if record matches filter
      record_no = recno()
      skip
      skip -1
      if recno() <> record_no
        go top
        if eof()
          * Remove filter
          filt_str = filter&num
          set filter to &filt_str
          filter_on&num = .F.
          @ 0,10 say space(9)
          go record_no
        endif
      endif
    endif (filter_on&num)
  else
    if .not. empty
      go record_no
    endif
  endif (choice = 'Y')
return

procedure skip
* Move forward/backward several records
  skip_no = 0
  @ 23,70 say '  Recs' get skip_no picture '@Z 999'
  read
  skip skip_no
  if eof()
    go bottom
    do load_var
    do disp_rec
    do disp_msg with 'Last Record'
  endif
  if bof()
```

```
    go top
    do load_var
    do disp_rec
    do disp_msg with 'First record'
  endif
return

procedure tally
* Count and display number of records in database
  @ 23,00
  @ 23,00 say 'Counting, please wait'
  count to number_
  do disp_msg with 'Count: ' + str (m->number_,6) + ' records'
  go record_no
return

* EOF ST006.PRG
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

**PROGRAM: ST008.PRG**

```
* Description:  Student Registration As Request
* Author:       Laddawan Chongsiriratanakul with GENIFER
* Date:         10/21/92    Time: 22:30
* Portions Copyright (c) 1986 Bytel Corporation. Use licensed by Bytel Corp.

* MAINTENANCE OPTIONS:
  * Duplic. keys: no
  * Modify key:   yes
  * Auto add:     yes
  * Partial key:  yes
  * Comments:     yes
  * Help screens: yes

PROCEDURE ST008

* FILES
  * Transaction File for Period
  database1 = 'STU_TRAN'
  index_fil1 = 'STU_TRAN'

* VARIABLES
  * Key in database STU_TRAN
    stu_id = 0
    term = ''
    mreg_date = date()

  * Fields in database STU_TRAN
    course_no = ''
    section_no = 0
    credit = 0
    tuition = 0

  * Flags
    abort_rec = .F.
    dupl_rec = .F.
    empty = .F.
    files_ok = .T.
    fld_ok = .T.
    conf_ok = .T.
    scr_ok = .T.
    valid_rec = .T.
    show_all = .T.
    filter_on1 = .F.
    set_filter = .F.

  * Others
    key1 = 'str(m->stu_id,10,0) + m->term'
    index1 = 'str(stu_id,10,0) + term'
    null_key1 = '         0'
    comp_key1 = 'trim(str(m->stu_id,10,0) + m->term)'
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
    keytran = 'str(m->stu_id,10,0) + x(k,2)'
    oldtran = 'str(old_id,10,0) + x(k,2)'
    keyprer = 'str(m->stu_id,10,0) + mprer'
    oldprer = 'str(old_id,10,0) + mprer'
    key4 = 'x(k,2) + str(x(k,3),1,0)'
    comp_fld  = 'stu_id=m->stu_id .and. term=m->term'
    key5 = 'substr(str(m->stu_id),3,3)'
    pak1 = .F.
    choice = ''
    option = ''
    filt_str = ''
    filter1 = ''
    scr_num = 1
    num = '1'
    max_screen = 1
    record_no = 0
    name = space(40)
    old_id = 0
    mprer = space(6)
    mamount = 0.00
    mcredit = 0
    mdisc = 0.00
    mtuit = 0.00
    mregis = 0.00
    mlibr = 0.00
    mclinic = 0.00
    mother1 = space(40)
    mother2 = space(40)
    mother3 = space(40)
    mother4 = space(40)
    mamt1 = 0.00
    mamt2 = 0.00
    mamt3 = 0.00
    mamt4 = 0.00
    mnet = 0.00
    runno = 1
    bntoc = space(3)
    cntoc = space(3)
    dntoc = space(3)
    xntoc = space(15)
    yntoc = space(15)
    zntoc = space(15)
    ntoc = space(50)

* Array Two Dimension
* Clear Array
   i = 1
   j = 1
   k = 1
   dimension x(8,6)
   do while i <= 8
      x(i,j) = 0
      x(i,j+1) = space(6)
      x(i,j+2) = 0
      x(i,j+3) = space(40)
```

```
      x(i,j+4) = 0
      x(i,j+5) = 0.00
      i = i+1
      j = 1
   enddo


* ENVIRONMENT
  set exact off
  set heading off
  set bell off
  set deleted on

* SCREEN
  do disp_scr

* OPEN FILES
  do chk_fils
  if .not. files_ok
    return
  endif
  index_str = index_fill
* select 1
* use stu_tran index stu_tran alias tran
  select 2
  use stu_sum index stu_sum alias summary
  select 3
  use stu_mas index stu_mas alias master
  select 4
  use stu_clas index stu_clas
  select 5
  use stu_plan index stu_plan

* PROCESSING LOOP
  do while valid_rec
     do init_key
     do init_fld
     do get_key
     read
     comp_key = comp_key&num
     if null_key&num = &comp_key
        * blank key
        valid_rec = .f.
        close all
        return
     endif
     select master
     seek str(m->stu_id,10,0)
     if .not. found()
        fld_ok = .f.
        do disp_msg with 'Master file not found'
     endif
     if fld_ok
        old_id = old_stu_id
        name = stu_name
```

```
@ 4, 30  say name
k=1
select 1
use stu_tran index stu_tran alias tran
seek &key1
do while &comp_fld
   x(k,1) = k
   x(k,2) = course_no
   x(k,3) = section_no
   x(k,5) = credit
   x(k,6) = tuition
   select 4
   seek &key4
   if found()
      x(k,4) = cour_title
   endif
   do fld_scn
   select tran
   skip
   k = k+1
enddo
select 1
use stu_tran index class_id,stu_term,stu_tran alias tran
go top
do while x(k,1) <> 99
   fld_ok = .t.
   do get_flds
   read
   if x(k,1) = 0
      close all
      return
   endif
   if x(k,1) <> 99
      if x(k,1) > 7
         fld_ok = .f.
         do disp_msg with 'Can not greater than 7 items'
      endif
      if fld_ok
         k=x(k,1)
         @ 19,19 say x(k,4)
         @ 19,61 say x(k,5) picture '9'
         @ 19,67 say x(k,6) picture '99999.99'
         @ 19, 7 get x(k,2)
         @ 19,17 get x(k,3) picture '9'
         read
      endif
      if x(k,2) = space(6) .and. fld_ok
         fld_ok = .f.
         do disp_msg with 'Course number is invalid'
      endif
      select 4
      seek &key4
      if .not. found() .and. fld_ok
         fld_ok = .f.
         do disp_msg with 'Course number is not found'
```

```
      else
         x(k,4) = cour_title
         x(k,5) = credit
         x(k,6) = tuition
         mprer  = prerequis
      endif
      select 1
      seek &keytran
      if found() .and. grade <> 'F ' .and. fld_ok
         fld_ok = .f.
         do disp_msg with 'This Course is already passed'
      endif
      if old_id <> 0 .and. fld_ok
         seek &oldtran
         if found() .and. grade <> 'F ' .and. fld_ok
            fld_ok = .f.
            do disp_msg with 'This Course is already passed'
         endif
      endif
      if mprer <> space(6) .and. fld_ok
         seek &keyprer
         if .not. found()
             seek &oldprer
             if .not. found()
                fld_ok = .f.
                do disp_msg with 'This Course should prerequite from number &mprer '
             endif
         endif
      endif
      if fld_ok
         @ 19,19 say x(k,4)
         @ 19,61 say x(k,5) picture '9'
         @ 19,67 say x(k,6) picture '99999.99'
         choice = 'Y'
         do get_optn with 'Confirm (Y/N)','YN',choice
         if choice = 'Y'
            do fld_scn
            k=k+1
         else
            x(k,4) = space(40)
            x(k,5) = 0
            x(k,6) = 0
         endif
         do clr_flds
         @ 23,00 say space(79)
      endif
   endif
enddo
k = 1
mamount = 0
mper = 0
mtuit = 0
mcredit = 0
do while x(k,2) <> space(6)
   mamount = mamount + x(k,6)
```

```
            mcredit = mcredit + x(k,5)
            k = k+1
         enddo
         mtuit = mamount - mper
         @ 19, 2 say space(60)
         set color to w+
         @ 17,44 to 21,78 double
         @ 18,46 say 'Total'
         @ 18,67 say mamount picture '999,999.99'
         @ 19,46 say 'Discount'
         set colo to u
         @ 19,67 say mper picture '999,999.99'
         set color to w+
         @ 20,46 say 'Net Amount'
         set color to w+
         @ 20,67 say mtuit picture '999,999.99'
         set color to w
         do while conf_ok
            @ 19,55 get mdisc picture '99.99 %'
            read
            mper = mamount*mdisc/100
            mtuit = mamount - mper
            @ 19,67 say mper picture '999,999.99'
            set color to w+
            @ 20,67 say mtuit picture '999,999.99'
            set color to w
            choice = 'Y'
            do get_optn with 'Confirm (Y/N)','YN',choice
            if choice = 'Y'
               conf_ok = .f.
               do screen
            endif
         enddo
      endif
      do disp_scr
   enddo
return

procedure fld_scn
* Show Array to Screen
   x(k,1) = k
   @ 9+k, 2 say x(k,1) picture '99.'
   @ 9+k, 7 say x(k,2)
   @ 9+k,17 say x(k,3) picture '9'
   @ 9+k,19 say x(k,4)
   @ 9+k,61 say x(k,5) picture '9'
   @ 9+k,67 say x(k,6) picture '99999.99'
return

procedure chk_fils
* Set files_ok to .F. and display a message if a file is missing;
* create an index file if one does not exist.
   close databases
   file_num = 1
   do while file_num <= max_screen .and. files_ok
```

```
seq = str(file_num,1)
if .not. file (database&seq + '.dbf')
  do disp_msg with 'File ' + database&seq + ' not found'
  files_ok = .F.
endif
if files_ok .and. .not. file (index_fil&seq + '.ntx')
  @ 23,00
  set color to w*
  @ 23,00 say 'Indexing'
  set color to w
  db = database&seq
  ind = index_fil&seq
  use &db
  ind_key = index&seq
  index on &ind_key to &ind
  use
endif
if files_ok .and. .not. file ('CLASS_ID.NTX')
  @ 23,00
  set color to w*
  @ 23,00 say 'Indexing'
  set color to w
  use stu_tran
  index on str(stu_id,10,0)+course_no to class_id
  use
endif
if files_ok .and. .not. file ('STU_TERM.NTX')
  @ 23,00
  set color to w*
  @ 23,00 say 'Indexing'
  set color to w
  use stu_tran
  index on term + course_no + str(section_no,1,0) to stu_term
  use
endif
if .not. file ('STU_MAS.DBF')
   do disp_msg with 'File STU_MAS not found'
   files_ok = .F.
endif
if files_ok .and. .not. file ('STU_MAS.NTX')
  @ 23,00
  set color to w*
  @ 23,00 say 'Indexing'
  set color to w
  use stu_mas
  index on str(stu_id,10,0) to stu_mas
  use
endif
if .not. file ('STU_CLAS.DBF')
   do disp_msg with 'File STU_CLAS not found'
   files_ok = .F.
endif
if files_ok .and. .not. file ('STU_CLAS.NTX')
  @ 23,00
  set color to w*
```

```
      @ 23,00 say 'Indexing'
      set color to w
      use stu_clas
      index on course_no + str(sect_no,1,0) to stu_clas
      use
    endif
    if .not. file ('STU_PLAN.DBF')
       do disp_msg with 'File STU_PLAN not found'
       files_ok = .F.
    endif
    if files_ok .and. .not. file ('STU_PLAN.NTX')
      @ 23,00
      set color to w*
      @ 23,00 say 'Indexing'
      set color to w
      use stu_plan
      index on str(plan_no,3,0) to stu_plan
      use
    endif
    if .not. file ('STU_SUM.DBF')
       do disp_msg with 'File STU_SUM not found'
       files_ok = .F.
    endif
    if files_ok .and. .not. file ('STU_SUM.NTX')
      @ 23,00
      set color to w*
      @ 23,00 say 'Indexing'
      set color to w
      use stu_sum
      index on str(stu_id,10,0) + term to stu_sum
      use
    endif

    file_num = file_num + 1
  enddo (while file_num <= max_screen .and. files_ok)
  @ 23,00
  @ 24,00
return

procedure clr_cal
* Clear computed screen areas
  @  3,18 say space (8)
  @ 19, 2 say space (8)
  @ 19,20 say space (8)
return

procedure clr_flds
* Clear screen field areas
  @ 19, 7 say space (6)
  @ 19,17 say space (1)
  @ 19,19 say space (40)
  @ 19,61 say space (1)
  @ 19,67 say space (8)
return
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIS VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
procedure disp_msg
parameters message
* Display "message" + '...' at line 23; wait for entry of any key
  @ 23,00
  @ 22,79
  wait message + '...'
return


procedure disp_scr
* Display stationary part of screen
  clear
  set color to n/r
  @  1,16 say 'I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)'
  set color to w
  @  1,72 say date()
  set color to w+
  @  2, 0 to   5,79 double
  @  3, 2 say 'Register Date :                    Term :'
  @  4, 2 say 'Student ID.    :'
  @  6, 0 to   9,79 double
  @  6, 0 to  22,79 double
  @  7, 2 say 'Item  Course  Section              Title'
  @  7,59 say 'Credit   Tuition'
  @  8, 8 say 'Number'
  @  8,70 say 'Fee'
  @ 21, 2 say '99-End of Job'
  set color to w
return


procedure screen
* Display stationary part of screen
    mregis = 0.00
    mlibr = 0.00
    mclinic = 0.00
    mother1 = space(40)
    mother2 = space(40)
    mother3 = space(40)
    mother4 = space(40)
    mamt1 = 0.00
    mamt2 = 0.00
    mamt3 = 0.00
    mamt4 = 0.00
    mnet = 0.00
    select 5
    seek &key5
    if found()
       mregis = regis
       mlibr = library
       mclinic = clinic
    endif
    mnet = mregis+mtuit+mlibr+mclinic
    clear
```

```
set color to n/r
@  1,16 say 'I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)'
set color to w
@  1,72 say date()
set color to w+
@  2, 0 to  22,79 double
@  3, 2 say 'Register Date :                    Term :'
set color to w
@  3,18 say mreg_date picture '@D'
@  3,41 say substr(m->term,1,2) + '/'+substr(m->term,3,2)
@  4,18 say m->stu_id picture '9999999999'
@  4,30 say name
set color to w+
@  4, 2 say 'Student ID.    :'
@  6, 2 to   8,77
@  6, 2 to  19,77
@  6,60 to  19,77
@  7,20 say 'Description'
@  7,65 say 'Amount'
@  9, 4 say 'Registration Fee'
@ 10, 4 say 'Tuition Fee'
@ 11, 4 say 'Library Fee'
@ 12, 4 say 'Physical Examination Or Clinic Fee'
@ 13, 4 say 'Other     1)'
@ 14, 4 say '          2)'
@ 15, 4 say '          3)'
@ 16, 4 say '          4)'
@ 18, 4 say 'The sum of '
set color to w
@ 18,15 say repl(chr(177),45)
@  9,66 say mregis  picture '999,999.99'
@ 10,66 say mtuit   picture '999,999.99'
@ 11,66 say mlibr   picture '999,999.99'
@ 12,66 say mclinic picture '999,999.99'
set color to w+
@ 18,66 say mnet    picture '999,999.99'
set color to w
do num_char with mnet
do while scr_ok
   @ 13,16 get mother1
   @ 13,66 get mamt1   picture '999,999.99'
   @ 14,16 get mother2
   @ 14,66 get mamt2   picture '999,999.99'
   @ 15,16 get mother3
   @ 15,66 get mamt3   picture '999,999.99'
   @ 16,16 get mother4
   @ 16,66 get mamt4   picture '999,999.99'
   read
   mnet = mregis+mtuit+mlibr+mclinic+mamt1+mamt2+mamt3+mamt4
   set color to w+
   @ 18,66 say mnet    picture '999,999.99'
   set color to w
   do num_char with mnet
   choice = 'Y'
   do get_optn with 'Confirm (Y/N)','YN',choice
```

UNIVERSITY OF THAILAND
มหาวิทยาลัยอัสสัมชัญ
SINCE 1969

```
        if choice = 'Y'
           scr_ok = .f.
           select 2
           seek &key1
           if .not. found()
              append blank
           endif
           replace stu_id with m->stu_id
           replace term with m->term
           replace reg_date with mreg_date
           replace amount with mamount
           replace credit with mcredit
           replace discount with mdisc
           replace per_dis with mper
           replace tuitionfee with mtuit
           replace regis with mregis
           replace library with mlibr
           replace clinic with mclinic
           replace other1 with mother1
           replace other2 with mother2
           replace other3 with mother3
           replace other4 with mother4
           replace amt1 with mamt1
           replace amt2 with mamt2
           replace amt3 with mamt3
           replace amt4 with mamt4
           replace net_amt with mnet
           k=1
           select 1
           do while x(k,2) <> space(6)
              seek &keytran
              if .not. found()
                 append blank
              endif
              replace stu_id with m->stu_id
              replace term with m->term
              replace course_no with x(k,2)
              replace section_no with x(k,3)
              replace credit with x(k,5)
              replace tuition with x(k,6)
              replace grade with space(2)
              k=k+1
            enddo
        endif
        @ 23,00 say space(70)
     enddo
return

procedure get_flds
* Get field variables
  x(k,1) = k
  @ 19, 2 get x(k,1) picture '99'
  @ 19, 4 say '.'
return
```

```
procedure get_key
* Get key variables
  @  3,18 get mreg_date picture '@D'
  @  3,41 get m->term
  @  4,18 get m->stu_id picture '9999999999'
return

procedure get_optn
parameters message, choices, choice
* Display the string "message" on line 23; get a character
* (defaulted to "choice"), validate it against "choices" and
* return in "choice" (must be a memory variable, not literal).
* If choice = 'A' display choice without accepting it.
  @ 23,00 say message + '? '
  @ 23,len(message)+3
  char = ' '
  do while .not. char $ choices
    char = choice
    @ 23,len(message) + 2 get char picture '!'
    if choice <> 'A'
      read
    else
      clear gets
    endif
  enddo (while .not. char $ choices)
  choice = char
return

procedure init_fld
* Clear field variables
  course_no = space(6)
  section_no = 0
  credit = 0
  tuition = 0.00
  i = 1
  j = 1
  k = 1
  do while i <= 8
     x(i,j) = 0
     x(i,j+1) = space(6)
     x(i,j+2) = 0
     x(i,j+3) = space(40)
     x(i,j+4) = 0
     x(i,j+5) = 0.00
     i = i+1
     j = 1
  enddo
return

procedure init_key
* Clear key variables
  mreg_date = date()
  term = space(4)
  stu_id = 0
  conf_ok = .t.
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
  scr_ok = .t.
  fld_ok = .t.
return

procedure num_char
parameter mnet
runno = 1
bntoc = substr(str(mnet,8,0),3,3)
cntoc = substr(str(mnet,8,0),6,3)
dntoc = bntoc
xntoc = space(15)
yntoc = space(15)
zntoc = space(15)
ntoc = space(50)
do while runno <=2
   do case
      case substr(dntoc,1,1) = '1'
           xntoc = 'One Hundred '
      case substr(dntoc,1,1) = '2'
           xntoc = 'Two Hundred '
      case substr(dntoc,1,1) = '3'
           xntoc = 'Three Hundred '
      case substr(dntoc,1,1) = '4'
           xntoc = 'Four Hundred '
      case substr(dntoc,1,1) = '5'
           xntoc = 'Five Hundred '
      case substr(dntoc,1,1) = '6'
           xntoc = 'Six Hundred '
      case substr(dntoc,1,1) = '7'
           xntoc = 'Seven Hundred '
      case substr(dntoc,1,1) = '8'
           xntoc = 'Eight Hundred '
      case substr(dntoc,1,1) = '9'
           xntoc = 'Nine Hundred '
   endcase
   do case
      case substr(dntoc,2,1) = '1'
           do case
              case substr(dntoc,3,1) = '0'
                   yntoc = 'Ten '
              case substr(dntoc,3,1) = '1'
                   yntoc = 'Eleven '
              case substr(dntoc,3,1) = '2'
                   yntoc = 'Twelve '
              case substr(dntoc,3,1) = '3'
                   yntoc = 'Thirteen '
              case substr(dntoc,3,1) = '4'
                   yntoc = 'Fourteen '
              case substr(dntoc,3,1) = '5'
                   yntoc = 'Fifty '
              case substr(dntoc,3,1) = '6'
                   yntoc = 'Sixteen '
              case substr(dntoc,3,1) = '7'
                   yntoc = 'Seventeen '
              case substr(dntoc,3,1) = '8'
```

```
                    yntoc = 'Eighteen '
                case substr(dntoc,3,1) = '9'
                    yntoc = 'Nineteen '
            endcase
        case substr(dntoc,2,1) = '2'
            yntoc = 'Twenty '
        case substr(dntoc,2,1) = '3'
            yntoc = 'Thirty '
        case substr(dntoc,2,1) = '4'
            yntoc = 'Forty '
        case substr(dntoc,2,1) = '5'
            yntoc = 'Fifty '
        case substr(dntoc,2,1) = '6'
            yntoc = 'Sixty '
        case substr(dntoc,2,1) = '7'
            yntoc = 'Seventy '
        case substr(dntoc,2,1) = '8'
            yntoc = 'Eighty '
        case substr(dntoc,2,1) = '9'
            yntoc = 'Ninety '
    endcase
    do case
        case substr(dntoc,3,1) = '1' .and. substr(dntoc,2,1) <> '1'
            zntoc = 'One '
        case substr(dntoc,3,1) = '2' .and. substr(dntoc,2,1) <> '1'
            zntoc = 'Two '
        case substr(dntoc,3,1) = '3' .and. substr(dntoc,2,1) <> '1'
            zntoc = 'Three '
        case substr(dntoc,3,1) = '4' .and. substr(dntoc,2,1) <> '1'
            zntoc = 'Four '
        case substr(dntoc,3,1) = '5' .and. substr(dntoc,2,1) <> '1'
            zntoc = 'Five '
        case substr(dntoc,3,1) = '6' .and. substr(dntoc,2,1) <> '1'
            zntoc = 'Six '
        case substr(dntoc,3,1) = '7' .and. substr(dntoc,2,1) <> '1'
            zntoc = 'Seven '
        case substr(dntoc,3,1) = '8' .and. substr(dntoc,2,1) <> '1'
            zntoc = 'Eight '
        case substr(dntoc,3,1) = '9' .and. substr(dntoc,2,1) <> '1'
            zntoc = 'Nine '
    endcase
    if runno = 1
        ntoc = xntoc+ltrim(yntoc)+ltrim(zntoc)
        if dntoc <> space(3)
            ntoc = xntoc+ltrim(yntoc)+ltrim(zntoc) + 'Thousand '
        endif
    else
        ntoc = ltrim(ntoc)+ltrim(xntoc)+ltrim(yntoc)+ltrim(zntoc) + 'Baht Only'
    endif
    xntoc = space(15)
    yntoc = space(15)
    zntoc = space(15)
    dntoc = cntoc
    runno = runno + 1
enddo
```

```
 @ 20,4 say space(73)
 @ 20,4 say ntoc
return

* EOF ST008.PRG
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

PROGRAM: ST102.PRG

```
* Description:  Student Information by Media Type
* Author:       Laddawan Chongsiriratanakul with GENIFER
* Date:         10/29/92    Time: 21:50
* Portions Copyright (c) 1986 Bytel Corporation. Use licensed by Bytel Corp.

PROCEDURE ST102

* FILES
  database = 'STU_MAS'
  index1 = 'STU_MAS'

* MEMORY VARIABLES
  page_no = 1
  line_no = 1
  record_no = 0
  top_marg = 6
  bot_marg = 6
  max_line = 66
  pg_ft_lins = 0
  outp_dev = ' '
  pr_ok = ' '
  new_sort = ' '
  files_ok = .T.
  ok = .F.
  break_on = .T.
  top_record = 0
  records = 0
  recs_on_pg = 0
  age = '(YEAR(DATE())-YEAR(STU_BIRTH))'

* ENVIRONMENT
  set heading off
  set scoreboard off
  set safety off
  set decimals to 10
  set margin to 0

* SCREEN
  clear
  @ 01,00 say 'ST102.PRG'
  @ 01,16 say 'STUDENT INFORMATION BY MEDIA'
  @ 01,54 say 'Primary File: ' + database

* OPEN FILES
  do chk_fils
  if .not. files_ok
    return
  endif
  select 1
  use &database alias database
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
* INDEX
  @ 04,00 say 'Special sort required?'
  do while .not. new_sort $ 'YN'
    new_sort = 'N'
    @ 04,24 get new_sort picture '!'
    read
  enddo
  if new_sort = 'Y'
    sort_exp = '' + space(60)
    legal_sort = .F.
    do while .not. legal_sort
      @ 06,00 say 'Index expression:' get sort_exp
      read
      if type(sort_exp) = 'U' .and. '' <> trim(sort_exp)
        do disp_msg with 'Undefined sort expression'
        @ 23,00
      else
        legal_sort = .T.
        if '' <> trim(sort_exp)
          index on &sort_exp to prim_ind
        endif
      endif
    enddo
  else
    if file(index1+'.ndx')
      set index to &index1
    endif
  endif (new sort = 'Y')

* DEVICE
  do while .not. ok
    @ 23,00
    @ 23,00 say 'Printer/Screen/Quit?'
    outp_dev = ' '
    do while .not. outp_dev $ 'PSQ'
      outp_dev = 'P'
      @ 23,21 get outp_dev picture '!'
      read
    enddo
    ok = .T.
    do case
      case outp_dev = 'P'
        @ 23,24 say 'Printer ready?'
        pr_ok = ' '
        do while .not. pr_ok $ 'YN'
          pr_ok = 'Y'
          @ 23,39 get pr_ok picture '!'
          read
        enddo
        if pr_ok = 'N'
          ok = .F.
        else
          @ 23,00 say 'Printing' + space(40)
          set device to print
```

```
          set console off
        endif
      case outp_dev = 'S'
        max_line = 23
        top_marg = 0
        bot_marg = 0
        clear
      case outp_dev = 'Q'
        close databases
        return
    endcase
  enddo (while .not. ok)

* REPORT HEADER
  go top
  record_no = recno()
  line_no = top_marg
  top_record = recno()
  recs_on_pg = 0
  do page_hd

* REPORT BODY
  do while .not. eof()
    record_no = recno()
    break_on = .F.
    do detail
    recs_on_pg = recs_on_pg + 1
    skip
  enddo (while .not. eof())

* REPORT FOOTER
  if outp_dev = 'S'
    @ 23,79
    wait 'Press any key'
  else
    eject
    set console on
    set device to screen
  endif (outp_dev = 'S')

  close databases
return

* PROCEDURES

procedure adv_line
  line_no = line_no + 1
  if line_no > max_line - bot_marg - pg_ft_lins
    do adv_page
  endif
return

procedure adv_page
  page_no = page_no + 1
  if outp_dev = 'S'
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
    @ 23,79
    wait 'Press any key'
    clear
  else
    eject
  endif
  top_record = recno()
  recs_on_pg = 0
  line_no = top_marg
  do page_hd
  line_no = line_no + 1
return

procedure chk_fils
* Return files_ok = .T. if all required files exist .F. otherwise.
* Create index file if necessary.
  close databases
  if .not. file (database + '.dbf')
    do disp_msg with 'File &database..DBF not found'
    files_ok = .F.
  endif
return

procedure detail
* Compute and display line of information
  do adv_line
  @ line_no,  1 say media_code picture '9'
  @ line_no,  6 say media_name
  @ line_no, 31 say &age picture '99'
  @ line_no, 38 say stu_sex
  @ line_no, 42 say stu_occu
  @ line_no, 62 say stu_title
return

procedure disp_msg
parameters message
  @ 23,00
  @ 22,79
  wait message + '...'
return

procedure page_hd
  * Print page header
  do adv_line
  @ line_no, 22 say 'IMAC (INFORMATION MANAGEMENT ACADEMIC CE'
  @ line_no, 62 say 'NNTRE)'
  do adv_line
  @ line_no, 26 say 'STUDENT  INFORMATION  BY  MEDIA  TYPE'
  do adv_line
  do adv_line
  @ line_no,  0 say '========================================'
  @ line_no, 40 say '========================================'
  do adv_line
  @ line_no,  4 say 'MEDIA'
  do adv_line
```

```
  @ line_no,  0 say 'TYPE   DESCRIPTION     PLAN#   AGE   SEX'
  @ line_no, 44 say 'OCCUPATION             TITLE'
  do adv_line
  @ line_no,  0 say '=========================================='
  @ line_no, 40 say '=========================================='
return

* EOF ST102.PRG
```

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

# APPENDIX G
# SCREEN DESIGN

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST. GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)        11/24/92
Raintree Office Garden, A3 Building FL4th
272 Soi Japanese School, Rama IX Rd. Bangkok 10310,Thailand

*** Computerized Information System ***

 1. Menu... System Setup

 2. Menu... Routine Operation

 3. Menu... Month-end and Year-end Works

 4. Menu... Reports as Request

 5. Menu... Inquiry as Request

 0. Exit

    Please Select Option : 0
```

Figure G-1 Screen Design of Main Menu

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)        11/24/92
Raintree Office Garden, A3 Building FL4th
272 Soi Japanese School, Rama IX Rd. Bangkok 10310,Thailand

                    *** Menu System Setup ***

                    1. Enter/Update Plan Table Information

                    2. Enter/Update Instructor Master Information

                    3. Enter/Update Student Class Information

                    4. Enter/Update Student Master Information

                    5. Enter/Update Prospect Master Information

                    0. Return to Main Menu

                        Please Select Option : 0
```

Figure G-2 Screen Design of Menu 'System Setup'

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 11/24/92

Plan No. 101

Plan Name. Diploma in Computer Studies (Eng-Ncc)

| Description | Amount |
|---|---|
| Registration Fee | 1,500.00 |
| Library Fee | 100.00 |
| Physical Examination or Clinic Fee | 100.00 |

Ret/Beg/End/Next/Prev/Skip/Modify/Add/Copy/Del/List/Filt/Tally/Help/Quit? M

Figure G-3 Screen Design of Plan Table Information

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
OMNIA
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 11/24/92

Instructor No. 1

Instructor Name Dr. Seri Sawatesenee

Ret/Beg/End/Next/Prev/Skip/Modify/Add/Copy/Del/List/Filt/Tally/Help/Quit? M

Figure G-4 Screen Design of Instructor Master Information

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
OMNIA
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 11/24/92

| | | | |
|---|---|---|---|
| Course No. | : 101101 | Section No. | : 1 |
| Course Title | : Computer Concept I | | |
| Duration (hours) | : 48 | Credit : 3 | Day : Thursday |
| Tuition (baht) | : 4000.00 | Time | : 18.00-21.00 PM. |
| Instructor No. | : 1 | Name | : Dr. Seri Sawatesenee |
| Rate/hr. | : 1000.00 | Baht. | |
| Prerequisite | : | | |

Ret/Beg/End/Next/Prev/Skip/Modify/Add/Copy/Del/List/Filt/Tally/Help/Quit? M

Figure G-5 Screen Design of Student Class Information

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
OMNIA
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
                I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)        11/24/92
┌──────────────────────────────────────────────────────────────────────────────┐
│ Student ID.      9210110001    Old Student ID.          0                    │
│ Name             Mr. Taveechai Prasitpaisal                                  │
│ Address          94  Pracharaj 1 Rd.                                         │
│                  Bangsue Bangkok                                Post Code  10800
│ Telephone                                                                    │
│ Marital Status   S <S-Single, M-Married, O-Other>    Sex M <M-Male, F-Female>
│ Date Of Birth    10/02/60    Nationality  Thai          Religion  Buddhism   │
│                                                                              │
│ Office Name      Thai Farmers Bank                                           │
│ Office Address   Silom Road                                                  │
│                                                                Post Code     │
│ Telephone        2701122                                                     │
│ Occupation       Employee                Title    Asst. Div. Mgr.            │
│ Media Code       4 <1-Magazine, 2-Television,3-Radio, 4-Other> Personnel Dept.
│                                                                              │
│ Contact Address  94  Pracharaj 1 Rd.                                         │
│                  Bangsue Bangkok                                Post Code  10800
│ Telephone                                                       Pros. #   000001
│ Training Job     Y  <Y,N>  Evaluation Grade A <A,B,C,D>         Recruited Y <Y,N>
└──────────────────────────────────────────────────────────────────────────────┘
Ret/Beg/End/Next/Prev/Skip/Modify/Add/Copy/Del/List/Filt/Tally/Help/Quit? M
```

Figure G-6 Screen Design of Student Master Information

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)

PROSPECT INFORMATION RUN BOOK

```
PROS.# 000001

Company Name : M.P. Marketing Co., Ltd. (Attn: Mr.Lek)
Telephone : 2493546                    Fax : 2497014
Contact Address : 1165-1177 Klongtuey
                  Prakanong, Bangkok 10110
Computer Environment :
          Hardware 1): System/36 Model 5362
                   2): PC
                   3):
          Software 1): BICARSA
                   2): PC Standalone
                   3):
Remark : AS/400 Planning Next Year (1993)
```

Ret/Beg/End/Next/Prev/Skip/Modify/Add/Copy/Del/List/Filt/Tally/Help/Quit? M

Figure G-7 Screen Design of Prospect Master Information

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)        12/01/92
Raintree Office Garden, A3 Building FL4th
272 Soi Japanese School, Rama IX Rd. Bangkok 10310,Thailand

*** Menu Routine Operation ***

1. Maintain Class Cost Control Analysis

2. Process Student Registration as Request

3. Process Student Payment as Request

4. Process Student Grade as Request


0. Return to Main Menu

Please Select Option : 0
```

Figure G-8 Screen Design of Menu 'Routine Operation'

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 11/24/92

| | | | | |
|---|---|---|---|---|
| Course No. 101101 | | | | |
| Duration | | | | 48 hr. |
| Instructor Fees 1: | 1,000.00 | 48,000.00 | | |
| 2: | 0.00 | 0.00 | | |
| 3: | 0.00 | 0.00 | | |
| 4: | 0.00 | 0.00 | | |
| 5: | 0.00 | 0.00 | | 48,000.00 Baht |
| Lab Cost | | | | 0.00 Baht |
| Student in Class 20 | Document Fee | | 0.00 | 0.00 Baht |
| Total Cost Control | | | | 48,000.00 Baht |
| Standard Mark-Up 40 % | | | | 19,200.00 Baht |
| Total Course Fee | | | | 67,200.00 Baht |
| Tuition per Person | | | | 3,360.00 Baht |
| Course Fee (Available) Mark-Up 20 % | | | | 9,600.00 Baht |
| Total Course Fee (Available) | | | | 57,600.00 Baht |
| Minimum Student (Available) | | | | 17 Unit |

Ret/Beg/End/Next/Prev/Skip/Modify/List/Filt/Tally/Help/Quit? R

Figure G-9 Screen Design of Class Cost Control Analysis

UNIVERSITY OF THAI CHAMBER OF COMMERCE
BROTHERS OF ST GABRIEL
OMNIA
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 11/24/92

| | |
|---|---|
| Register Date : 11/24/92 | Term : |
| Student ID. : 0 | |

| Item | Course Number | Section | Title | Credit | Tuition Fee |
|---|---|---|---|---|---|
| | | | | | |
| 99-End of Job | | | | | |

Figure G-10 Screen Design of Student Registration (Screen 1)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 11/24/92

Register Date : 11/24/92 Term : 9201
Student ID. : 9210110001 Mr. Taveechai Prasitpaisal

| Item | Course Number | Section | Title | Credit | Tuition Fee |
|---|---|---|---|---|---|
| 1. | 101101 | 1 | Computer Concept I | 3 | 4000.00 |
| 2. | 101102 | 1 | Computer Programming | 3 | 4000.00 |
| 3. | 101105 | 1 | Business Organization | 3 | 2000.00 |
| 4. | 101110 | 1 | Use of Application Packages | 3 | 2000.00 |
| 5. | | | | | |

99-End of Job

Figure G-11 Screen Design of Student Registration (Screen 2)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST. GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 11/24/92

Register Date : 11/24/92 Term : 9201
Student ID. : 9210110001 Mr. Taveechai Prasitpaisal

| Item | Course Number | Section | Title | Credit | Tuition Fee |
|---|---|---|---|---|---|
| 1. | 101101 | 1 | Computer Concept I | 3 | 4000.00 |
| 2. | 101102 | 1 | Computer Programming | 3 | 4000.00 |
| 3. | 101105 | 1 | Business Organization | 3 | 2000.00 |
| 4. | 101110 | 1 | Use of Application Packages | 3 | 2000.00 |

| | |
|---|---|
| Total | 12,000.00 |
| Discount 0.00 % | 0.00 |
| Net Amount | 12,000.00 |

99-End of Job

Figure G-12 Screen Design of Student Registration (Screen 3)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 11/24/92

Register Date : 11/24/92 Term : 9201
Student ID. : 9210110001 Mr. Taveechai Prasitpaisal

| Item | Course Number | Section | Title | Credit | Tuition Fee |
|---|---|---|---|---|---|
| 1. | 101101 | 1 | Computer Concept I | 3 | 4000.00 |
| 2. | 101102 | 1 | Computer Programming | 3 | 4000.00 |
| 3. | 101105 | 1 | Business Organization | 3 | 2000.00 |
| 4. | 101110 | 1 | Use of Application Packages | 3 | 2000.00 |

| | | |
|---|---|---|
| Total | | 12,000.00 |
| Discount | 5.00 % | 600.00 |
| Net Amount | | 11,400.00 |

99-End of Job

Confirm (Y/N)? Y

Figure G-13 Screen Design of Student Registration (Screen 4)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
OMNIA
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 11/24/92

Register Date : 11/24/92 Term : 92/01
Student ID. : 9210110001 Mr. Taveechai Prasitpaisal

| Description | Amount |
|---|---|
| Registration Fee | 1,500.00 |
| Tuition Fee | 11,400.00 |
| Library Fee | 100.00 |
| Physical Examination or Clinic Fee | 100.00 |
| Other 1) | 0.00 |
| 2) | 0.00 |
| 3) | 0.00 |
| 4) | 0.00 |
| The sum of | 13,100.00 |

Thirteen Thousand One Hundred Baht Only

Figure G-14 Screen Design of Student Registration (Screen 5)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST. GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 11/24/92

Register Date : 11/24/92 Term : 92/01
Student ID. : 9210110001 Mr. Taveechai Prasitpaisal

| Description | Amount |
|---|---|
| Registration Fee | 1,500.00 |
| Tuition Fee | 11,400.00 |
| Library Fee | 100.00 |
| Physical Examination or Clinic Fee | 100.00 |
| Other 1) Placement Test | 100.00 |
| 2) Insurance | 500.00 |
| 3) Activities | 1,000.00 |
| 4) | 0.00 |
| The sum of | 14,700.00 |

Fourteen Thousand Seven Hundred Baht Only

Confirm (Y/N)? Y

Figure G-15 Screen Design of Student Registration (Screen 6)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 10/22/92

Student ID. : 9210110001 Term : 9201 Mr. Taveechai Prasitpaisal
Payment Date: 10/22/92

| Description | Amount |
|---|---|
| Registration Fee | 1,500.00 |
| Tuition Fee | 11,400.00 |
| Library Fee | 100.00 |
| Physical Examination or Clinic Fee | 100.00 |
| Other 1) Placement Test | 100.00 |
| 2) Insurance | 500.00 |
| 3) Activities | 1,000.00 |
| 4) | 0.00 |
| The sum of | 14,700.00 |

[ ] Cash
[ ] Cheque Bank No.

Ret/Beg/End/Next/Prev/Skip/Modify/Add/Copy/Del/List/Filt/Tally/Help/Quit? M

Figure G-16 Screen Design of Student Payment (Screen 1)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 10/22/92

Student ID. : 9210110001 Term : 9201 Mr. Taveechai Prasitpaisal
Payment Date: 10/22/92

| Description | Amount |
|---|---|
| Registration Fee | 1,500.00 |
| Tuition Fee | 11,400.00 |
| Library Fee | 100.00 |
| Physical Examination or Clinic Fee | 100.00 |
| Other 1) Placement Test | 100.00 |
| 2) Insurance | 500.00 |
| 3) Activities | 1,000.00 |
| 4) | 0.00 |
| The sum of | 14,700.00 |

[ ] Cash
[X] Cheque Bank = Thai Farmers Bank, Head Office No. 21222009

Save (Y/N)? Y

Figure G-17 Screen Design of Student Payment (Screen 2)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 10/23/92

Term 9201 Course numbers 101101 Section 1

| Student ID. | Name | Credit | Semester Score | Grade | G.P.A. |
|---|---|---|---|---|---|
| | | | | | |

Figure G-18 Screen Design of Student Grade (Screen 1)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
OMNIA
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 10/23/92

Term 9201 Course numbers 101101 Section 1

| Student ID. | Name | Credit | Semester Score | Grade | G.P.A. |
|---|---|---|---|---|---|
| 1. 9210110001 | Mr. Taveechai Prasitpaisal | 3 | 0.00 | | 0.00 |
| 2. 9210110002 | Mr. Chansak Assawasirisilp | 3 | 0.00 | | 0.00 |
| 3. 9210110003 | Miss Jarumas Orvichian | 3 | 0.00 | | 0.00 |
| 4. 9210110004 | Miss Srisuda Tantimongkolsuk | 3 | 0.00 | | 0.00 |
| 5. 9210110005 | Mr. Preecha Jintasaerewonge | 3 | 0.00 | | 0.00 |
| 6. 9210110006 | Miss Suwimon Techasopongmanee | 3 | 0.00 | | 0.00 |
| 7. 9210110008 | Mr. Visilp Vongvanichvathana | 3 | 0.00 | | 0.00 |
| 8. 9210110009 | Mrs. Anong Nawajongpand | 3 | 0.00 | | 0.00 |

Figure G-19 Screen Design of Student Grade (Screen 2)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 10/23/92

Term 9201 Course numbers 101101 Section 1

| Student ID. | Name | Credit | Semester Score | Grade | G.P.A. |
|---|---|---|---|---|---|
| 1. 9210110001 | Mr. Taveechai Prasitpaisal | 3 | 75.00 | B | 3.00 |
| 2. 9210110002 | Mr. Chansak Assawasirisilp | 3 | 87.00 | A- | 3.70 |
| 3. 9210110003 | Miss Jarumas Orvichian | 3 | 68.00 | C+ | 2.30 |
| 4. 9210110004 | Miss Srisuda Tantimongkolsuk | 3 | 91.00 | A | 4.00 |
| 5. 9210110005 | Mr. Preecha Jintasaerewonge | 3 | 73.00 | B- | 2.70 |
| 6. 9210110006 | Miss Suwimon Techasopongmanee | 3 | 62.00 | C | 2.00 |
| 7. 9210110008 | Mr. Visilp Vongvanichvathana | 3 | 83.00 | B+ | 3.30 |
| 8. 9210110009 | Mrs. Anong Nawajongpand | 3 | 59.00 | F | 0.00 |

Confirm (Y/N)? Y

Figure G-20 Screen Design of Student Grade (Screen 3)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
OMNIA
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)        12/01/92
Raintree Office Garden, A3 Building FL4th
272 Soi Japanese School, Rama IX Rd. Bangkok 10310,Thailand

*** Menu Month-end and Year-end Works ***

 1. Backup before Generate Student History

 2. Post Student History Information

 3. Backup after Generate Student History

 4. Chop Student History Information


 0. Return to Main Menu

    Please Select Option : 0
```

Figure G-21 Screen Design of Menu 'Month-end and Year-end works'

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 12/01/92

| Term | Course numbers | Section 0 |
|---|---|---|

| Post Student History File |
|---|
| Total Student in Class |

Figure G-22 Screen Design of Post Student History (Screen 1)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE) 12/01/92

Term 9201 Course numbers 101101 Section 1

Post Student History File

Total Student in Class 8

Post Already.....

Figure G-23 Screen Design of Post Student History (Screen 2)

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
OMNIA
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)        12/01/92
Raintree Office Garden, A3 Building FL4th
272 Soi Japanese School, Rama IX Rd. Bangkok 10310,Thailand

*** Menu Reports as Request ***

 1. Student Name & Address Run Book      11. Prospect Information Run Book
 2. Student Information by Media Type    12. Student Payment Report
 3. Class-offering Report                13. Course Schedule Listing
 4. Cost Control Analysis Report         14. Grade Report for Student
 5. Student Outlet by Course No.         15. Student's Transcript as Request
 6. Student Registration in Each Term    16. Student's Certificate as Request
 7. Tuition Fee for Each Course          17. Class Roster for Instructor
 8. Student Movement Infor. in Each Year18. Student Grade for Instructor
 9. Student Progress Summary Report      19.
10. Top Ranking Analysis by Area Code    20.
                                          0. Return to Main Menu

                Please Select Option : 0
```

Figure G-24 Screen Design of Menu 'Reports as Request'

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

```
I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)        12/01/92
Raintree Office Garden, A3 Building FL4th
272 Soi Japanese School, Rama IX Rd. Bangkok 10310,Thailand

                *** Menu Inquiry as Request ***

                1. Student Information by Student ID
                2. Student Information by Student Name
                3. Course Schedule by Plan No.
                4.
                5.
                6.
                7.
                8.
                9.
               10.
                0. Return to Main Menu

                   Please Select Option : 0
```

Figure G-25 Screen Design of Menu 'Inquiry as Request'

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

# APPENDIX H
# OUTPUT REPORTS

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST. GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)
CLASS ROSTER
TERM ID 01/92

PLAN ID 101 PLAN NAME Diploma in Computer Studies (Eng-NCC)

COURSE# 101101 COURSE NAME Computer Concept I SEC. NO. 1 CREDIT HOURS 48

INSTRUCTOR NAME Mrs. Supaporn T. DAYS-OFFERED Thursday TIME-OFFERED 18.00 - 21.00

| NO. | STUDENT ID. | STUDENT NAME |
|---|---|---|
| 01 | 9210110001 | Mr. Taveechai Prasitpaisal |
| 02 | 9210110002 | Mr. Chansak Assawasirisilp |
| 03 | 9210110003 | Miss Jarumas Orvichian |
| 04 | 9210110004 | Miss Srisuda Tantiwongkolsuk |
| 05 | 9210110005 | Mr. Preecha Jintasaerewonge |
| 06 | 9210110006 | Miss Suwimon Techasopongmanee |
| 07 | | |
| 08 | | |
| 09 | | |
| 10 | | |
| 11 | | |
| 12 | | |
| 13 | | |
| 14 | | |
| 15 | | |
| 16 | | |
| 17 | | |
| 18 | | |
| 19 | | |
| 20 | | |
| 21 | | |
| 22 | | |
| 23 | | |
| 24 | | |

6 *** TOTAL STUDENTS ***

FIGURE H-1 DESIGN OF CLASS ROSTER OUTPUT FORM

# INFORMATION MANAGEMENT ACADEMIC CENTRE
# TRANSCRIPT

STUDENT ID : 9210110001

NAME : Mr. Taveechai Prasitpaisal

ADDR : 94 Pracharaj 1 Rd.
Bangsue Bangkok

PLAN : Diploma in Computer Studies

| COURSE NO. | COURSE TITLE | CR. | GR. |
|---|---|---|---|
| TERM ID. (YY/SM) - 92/01 | | | |
| 101101 | Computer Concept I | 3 | X |
| 101102 | Computer Programming | 3 | X |
| 101105 | Business Organisation | 3 | X |
| 101110 | Use of Appli. Packages | 3 | X |
| SEMESTER | 192 HA. 12 CR. 0.00 GPA. | | |
| CUMULATIVE | 192 HA. 12 CR. 0.00 GPA. | | |

| COURSE NO. | COURSE TITLE | CR. | GR. |
|---|---|---|---|
| TERM ID. (YY/SM) - 92/02 | | | |
| 101201 | Computer Concept II | 3 | X |
| 101205 | Information Management | 3 | X |
| SEMESTER | 999 HA. 99 CR. 9.99 GPA. | | |
| CUMULATIVE | 999 HA. 99 CR. 9.99 GPA. | | |
| TERM ID. (YY/SM) - 99/99 | | | |
| XXXXXX | XXXXXXXXXXXXXXXXXXXX | 9 | X |
| XXXXXX | XXXXXXXXXXXXXXXXXXXX | 9 | X |
| XXXXXX | XXXXXXXXXXXXXXXXXXXX | 9 | X |
| XXXXXX | XXXXXXXXXXXXXXXXXXXX | 9 | X |
| SEMESTER | 999 HA. 99 CR. 9.99 GPA. | | |
| CUMULATIVE | 999 HA. 99 CR. 9.99 GPA. | | |
| TERM ID. (YY/SM) - 99/99 | | | |
| XXXXXX | XXXXXXXXXXXXXXXXXXXX | 9 | X |
| XXXXXX | XXXXXXXXXXXXXXXXXXXX | 9 | X |
| XXXXXX | XXXXXXXXXXXXXXXXXXXX | 9 | X |
| XXXXXX | XXXXXXXXXXXXXXXXXXXX | 9 | X |
| SEMESTER | 999 HA. 99 CR. 9.99 GPA. | | |
| CUMULATIVE | 999 HA. 99 CR. 9.99 GPA. | | |

CERTIFIED TRUE COPY

REGISTRAR

EDUCATION MANAGER

DATE : ( 99 / 99 / 99 )

*** NOT VALID WITHOUT SEAL ***

FIGURE H-2 DESIGN OF STUDENT TRANSCRIPT OUTPUT FORM

IMAC (INFORMATION MANAGEMENT ACADEMIC CENTRE)

PLAN : Diploma in Computer Studies (Eng-NCC)

C O U R S E  S C H E D U L E  FOR  92 / 01

| COURSE# | COURSE TITLE | SEC# | HRS. | DAY | TIME | INST. NAME |
|---|---|---|---|---|---|---|
| 101101 | Computer Concept I | 1 | 48 | Thursday | 18.00 - 21.00 | Mrs.Supaporn Tummasiri |
| 101102 | Computer Programming | 1 | 48 | Friday | 18.00 - 21.00 | Dr. Seri Savatesenee |
| 101105 | Business Organisation | 1 | 48 | Saturday | 13.00 - 16.00 | Ms. Pornpak Ngamdee |

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

FIGURE H-3 DESIGN OF COURSE SCHEDULE OUTPUT FORM

IMAC (INFORMATION MANAGEMENT ACADEMIC CENTRE)

C L A S S  C O S T  C O N T R O L  A N A L Y S I S

| COURSE# | COURSE TITLE | TOTAL COST CONTROL | MK-UP (%) | ST. IN CLASS | FEES/ ST. | OPEN MK-UP(%) | AT LEAST STUDENT |
|---|---|---|---|---|---|---|---|
| 302001 | AS/400 System Operation | 20,000.00 | 30 | 13 | 2,000.00 | 10 | 11 |
| 302002 | AS/400 RDB Design&Cod. | 26,000.00 | 30 | 13 | 2,600.00 | 10 | 11 |
| 302004 | RPG Programm. for Beginner | 34,000.00 | 40 | 10 | 4,800.00 | 5 | 7 |

FIGURE H-4 DESIGN OF CLASS COST CONTROL REPORT

IMAC (INFORMATION MANAGEMENT ACADEMIC CENTRE)

STUDENT NAYE & ADDRESS RUN BOOK

FOR THE DATE OF 31/10/92

| PLAN NAME | NO. | STUDENT ID | STUDENT NAME | STUDENT ADDRESS | TEL. |
|---|---|---|---|---|---|
| Diploma in Computer Studies (Eng-NCC) | 101 | 9210110001 | Mr. Taveechai Prasitpai | 94 Prachraj 1 Rd. Bangsue, Bangkok 10000 | 585-1426 |
| | 101 | 9210110002 | Mr. Chansak Assawasiri | 14/22-25 Rongmuang Prathumwan, Bangkok 10330 | 214-2273 |
| Short Course in Computer Studies (Thai-IMAC) | 302 | 9230210001 | Ms. Darika Yoosaward | 39 St. Louis Soi 3 Yannawa, Bangkok 10120 | 212-2348 |
| | 302 | 9230210002 | Miss Vatinee Temponsub | 98 Kasemraj road Prakhanong, Bangkok 10110 | 249-3467 |

FIGURE H-5 DESIGN OF STUDENT NAME & ADDRESS RUN BOOK

I M A C (INFORMATION MANAGEMENT ACADEMIC CENTRE)
STUDENT GRADE REPORT
TERM ID 92/01

PLAN ID 101 PLAN NAME Diploma in Computer Studies (Eng-NCC)

COURSE# 101101 COURSE NAME Computer Concept I SEC. NO. 1 CREDIT HOURS 48

INSTRUCTOR NAME Mrs. Supaporn T. DAYS-OFFERED Thursday TIME-OFFERED 18.00 - 21.00

| NO. | STUDENT ID. | STUDENT NAME | CREDIT | SCORE | GRADE | GPA. |
|---|---|---|---|---|---|---|
| 01 | 9210110001 | Mr. Taveechai Prasitpaisal | 3 | 75 | B | 3.00 |
| 02 | 9210110002 | Mr. Chansak Assawasirisilp | 3 | 87 | A- | 3.70 |
| 03 | 9210110003 | Miss Jarumas Orvichian | 3 | 68 | C+ | 2.30 |
| 04 | 9210110004 | Miss Srisuda Tantimongkolsuk | 3 | 91 | A | 4.00 |
| 05 | 9210110005 | Mr. Preecha Jintasaerewonge | 3 | 73 | B- | 2.70 |
| 06 | 9210110006 | Miss Suwimon Techasopongmanee | 3 | 62 | C | 2.00 |

OPERATED BY _____ APPROVED BY _____

REGISTRAR INSTRUCTOR

FIGURE H-6 DESIGN OF STUDENT GRADE REPORT

IMAC (INFORMATION MANAGEMENT ACADEMIC CENTRE)

STUDENT INFORMATION BY MEDIA TYPE

AT 30 / 10 / 92

| MEDIA TYPE | MEDIA DESCRIPTION | PLAN # | AGE | SEX | OCCUPATION | TITLE |
|---|---|---|---|---|---|---|
| 1 | Computer Reviews | 101 | 28 | M | Employee | Programmer |
| 4 | Personnel Dept. | 101 | 30 | M | Employee | Asst. Div. Mgr. |
| 1 | Computer Reviews | 302 | 25 | M | Employee | Programmer |
| 1 | Computer Reviews | 302 | 27 | M | Employee | System Operator |
| 1 | Computer Today | 302 | 29 | F | Owner | Manager |

FIGURE H-7 DESIGN OF STUDENT INFORMATION SEPARATED BY MEDIA TYPE

I M A C
PROSPECT INFORMATION RUN BOOK
AT THE END OF 15/11/92

| PROS.# | CONTACT NAME | CONTACT ADDRESS | TEL/FAX | <-COMPUTER ENVIRONMENT-> H/W TYPE | S/W TYPE |
|---|---|---|---|---|---|
| 920001 | Ms.Supannee Jeksakul | MP.Marketing Co.,Ltd. | 249-3546 | 1. S/36 Mod. 5362 | 1. BICARSA |
| | | 1165-1177 Klongtuey | 249-7014 | 2. PC | |
| | | Prakanong, Bangkok 10110 | | | |
| | | Remark : AS/400 Planning for next year (1993) | | | |
| 920002 | Mr.Somnuek Taicha | ABC Insurance Company | 231-0900 | 1. AS/400 Mod. D45 | 1. RPG III |
| | | 15 Silom Road, | 231-5005 | 2. LAN | 2. UNIX |
| | | Bangrak, Bangkok 10500 | | | |

FIGURE H-8 DESIGN OF PROSPECT INFORMATION RUN BOOK

IMAC (INFORMATION MANAGEMENT ACADEMIC CENTRE)

C L A S S - O F F E R I N G  R E P O R T

AT THE END OF 11 / 92

| PLAN NAME | COURSE# | COURSE TITLE | SEC-HRS | DAY | TIME | INST. NAME |
|---|---|---|---|---|---|---|
| Diploma in | - 101101 | Computer Concept I | 1-48 | Thursday | 18.00 - 21.00 | Mrs.Supaporn Tummasiri |
| Computer | - 101102 | Computer Programming | 1-48 | Friday | 18.00 - 21.00 | Dr. Seri Savatesenee |
| Studies | - 101105 | Business Organisation | 1-48 | Saturday | 13.00 - 16.00 | Ms. Pornpak Ngamdee |
| Short Course | - 302001 | AS/400 System Operation | 1-12 | 1-2 Nov. | 09.00 - 16.00 | Mr. Somchai Taechaidej |
| in Computer | - 302001 | AS/400 System Operation | 2-12 | 3-4 Nov. | 09.00 - 16.00 | Mr. Somchai Taechaidej |
| Studies | - 302002 | AS/400 RDB Design&Cod. | 1-18 | 6-8 Nov. | 09.00 - 16.00 | Ms. Lakana Noipricha |

FIGURE H-9 DESIGN OF CLASS-OFFERING REPORT

IMAC (INFORMATION MANAGEMENT ACADEMIC CENTRE)

TUITION FEE FOR EACH COURSE SCHEDULE

FOR THE YEAR 1992

| PLAN | COURSE# | COURSE TITLE | SEC | HRS | DAY | TIME | FEES |
|---|---|---|---|---|---|---|---|
| Diploma in Computer | 101101 | Computer Concept I | 1 | 48 | Thursday | 18.00 - 21.00 | 4,000 |
| Studies (Eng-MCC) | 101102 | Computer Programming | 1 | 48 | Friday | 18.00 - 21.00 | 4,000 |
| | 101105 | Business Organisation | 1 | 48 | Saturday | 13.00 - 16.00 | 4,000 |
| Short Course in | 302001 | AS/400 System Operation | 1 | 12 | 1-2 Nov. | 09.00 - 16.00 | 2,000 |
| Computer Studies | 302001 | AS/400 System Operation | 2 | 12 | 3-4 Nov. | 09.00 - 16.00 | 2,000 |
| (Thai-IMAC) | 302002 | AS/400 RDB Design&Cod. | 1 | 18 | 6-8 Nov. | 09.00 - 16.00 | 2,600 |

FIGURE H-10 DESIGN OF TUITION FEE FOR EACH COURSE SCHEDULE

ABAC
GRADUATE SCHOOL LIBRARY

IMAC (INFORMATION MANAGEMENT ACADEMIC CENTRE)

STUDENT OUTLET BY COURSE NO.

AT 30 / 10 / 92

| PLAN # | COURSE # | SEC# | COURSE NAME | STATUS M | STATUS F | TOTAL STUDENTS |
|---|---|---|---|---|---|---|
| 101 | 101101 | 1 | Computer Concept I | 8 | 10 | 18 |
| 101 | 101102 | 1 | Computer Programming | 8 | 9 | 17 |
| 302 | 302001 | 1 | AS/400 System Operation | 10 | 12 | 22 |
| 302 | 302002 | 1 | AS/400 RDB Design&Cod. | 8 | 7 | 15 |
| 302 | 302003 | 1 | AS/400 Application Programming | 9 | 10 | 19 |

ASSUMPTION UNIVERSITY OF THAILAND
BROTHERS OF ST GABRIEL
LABOR OMNIA VINCIT
SINCE1969
มหาวิทยาลัยอัสสัมชัญ

FIGURE H-11 DESIGN OF STUDENT OUTLET BY COURSE NO.