

ADDIS **ABABA SCIENCE AND TECHNOLOGY UNIVERSITY**

COLLEGE ELECTRICAL AND MECHANICAL ENGINEERING

DEPARTEMENT OF SOFTWARE ENGINEERING

FOP II Project Documentation

Name ID

Elsabeth Kahsay 0241/12

Elsahaday Dereje 0243/12

Eyobed Mesfin 0271/12

Natan Mekebib 1029/12

Submitted to Mr. Chere Lemma

Submission Date Sept 23 2021

Page | 0

Table of contents

Table of Contents

1. Introduction3
   1. Overview3
   2. General Objectives3
   3. Specific Objectives3
   4. Significance4
   5. Limitations 5
2. System and development life cycle4
   1. Requirement Analysis7
      1. Software Requirements 7
      2. Hardware performance and requirements7
3. System and development life cycle4
   1. Structures Used8
   2. Functions Used9
   3. Flow Chart14

Abstract

The purpose of this program of school billing system is to assist schools with necessary technical management operations like bill management, registration, statistical reports and many more. This code will help to calculate student’s fee, school’s expenses and profit with ease than doing it manually using paper and pen. This will mostly help those schools who lack proper file management systems and software implementations.

1. **Introduction**
   1. Overview

This school management code can register, Update, Delete student information, produces receipt after payment and in addition to that gives multiple admin options for an authorized figure like setting course payment and intake capacity for a particular year and also produce various statistical reports in tabular format as per required. Options mentioned above and others help to calculate things like student fee, school expense and profit with the help of automated system which not only saves time and energy but also efficiently.

* 1. General objectives

This project’s generally aims to assist schools to have efficient, well organized, automatic billing system that records and saves information accurately.

* 1. Specific Objective

Some of this project’s specific objectives are the following;

* Register, Update, Delete student information and produce receipt in a time saving and efficient way
* Give multiple Admin options, which are password protected, for a better management and organization of the school system
* Calculate fees, expenses and profits accurately and easily
* Summarize and present statistical information as per requested
  1. Significance

As mentioned, multiple times this project is very useful in various ways

1. Significance to Parents/ legal Guardians of a student

* Easy to use and user-friendly interface
* To check payment status online
* Check enrollment availability along with the courses easily
* Search and edit option

1. To the management

* To easily set up necessary information about student registration, intake capacity and fees.
* User friendly interface
* To quickly access school expenses and profits
* Easy access to statistical information
  1. Limitations
* The code can only display limited number of courses.
* Full name of student must be entered for proper saving of record
* Payment options is not yet available so user has no direct payment method
* Editing requires changing all record entries

Concepts implemented:

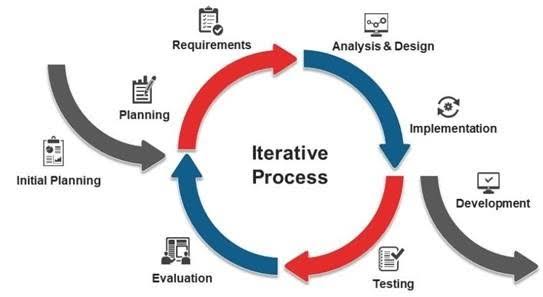
* Pointer
* Structure
* Dynamic allocation (Phase I)
* Vector container (Phase II)

1. **System Development Life Cycle**

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process. Which includes Planning, defining, designing, building, testing and deploying.

In this project we used one of the many SDLC models namely Iterative Model.

An iterative life cycle model is a type of model which doesn’t attempt to start with a full specification of requirements hence why we chose it. Its development begins by specifying and implementing just part of the software, which later can be reviewed and edited producing a new version. This model has many advantages like testing and debugging are easier during smaller iteration. Risk analysis is also way better.



* 1. Requirement Analysis

**2.1.1 Software requirements**

The system should fulfil the following requirements

* Reliable: since part of this code’s job is calculating expenses, fees and profits it should be reliable and accurate
* User friendly: the system should be clear and easy to understand for the user to use
* Security: The system should be secure especially the administration option, should be password protected from unauthorized person access

**2.1.2 Hardware performance and requirements**

Program Performance:

* 0% CPU Processing
* 7.8 MBs of RAM
* 0MB/S data transmission to the Internal Drive
* And very low power usage

Minimum recommendations:

* At least 230 KBs of free space on internal memory
* At least 2 GB of RAM memory
* On any Core I Intel Computers

1. **System Overview**

As mentioned, this system can register, update, delete, alter and displays tabular statistical reports receipts.

Structures and functions are used to effective handle stored inputs and records.

A Structure is a collection or aggregate container of related data items, possibly of different types and is user defined data type.

* 1. Structures used to handle records are

1. struct Courselist

This structure contains course names with string data type and price with float data type

1. struct Date

This structure contains only date, month and year represented as dd.mm,yy with int as their data type.

1. struct studInfo

This structure contains more variable than the previous two and also contains nested structure. It includes name, last name and sex with char as their data type. Age and student ID with int data type. Regestration date and pay date with the date data type which is user defined. Courses with course list as data type which is also user defined. And lastly total pay with float data type.

1. struct Schoolinfo

This structure contains total, expense and net with float data type.

function is block of instruction that is executed when it is called form some other point of a program. They can be compiled and tested separately and reused in different programs.

3.2 Functions used to perform different operations in the system are

1. void new\_reg()

This function

* Displays the user menu and allows the user to choose from the 6 choices and calls 5 functions.
* It allows the user to register N-amount of students and it compares that number to the number of available spots left(vacancies) and if it's allowed it will proceed to register a new student
* Checks payment and appends it to the students name
* Produces receipt by collecting info from the registration
* It allows to enter different data types and personal details of students(like names & age)
* writes the input (student info ) to ID.txt text file.
* It has options to go back to the main menu

2. void admin()

* Sets password for the admin option to grant access and will exit the system if more than 3 wrong attempt
* Allows the user to enter a choice out 3 options in the admin menu using switch and cases and calls their respective functions.

3.void set\_intake()

* Allows the admin to set new intake capacity
* Compares student count and intake capacity to display a message

4. void set\_pay()

* Allows the admin to set new price for number of courses
* Has a retrial option

1. void overdue()

* Compares todays date with the pay date
* generates individual table elements
* Has a retrial option

1. void profit()

* Calculates school total expenses and school net using the provided algorithm
* Displays the calculated expense and profit
* Has a retrial option

1. void Coursechoice(int studnum)

* Allows the user to enter the number of course to be taken by displaying the course menu
* Accepts choices using arrays
* Calculates total pay using arrays

1. inline void table(int i)

* displays individual table elements to the user info including ID, name, age, sex, pay date and total payment
* It accepts an integer I which is the reference index of a student object

1. void reorder()

* Is used to swap structures for the purpose of ordering records by name alphabetically.

1. void statreport()

* Allows the user to choose by displaying the Statistics and Report Menu using switch and cases
* Displays tabular form report including name, age, sex, payment date and total payment
* Displays report based on the choice chosen by the user

1. void searching()

* Gives the option of searching by name or ID of the student
* Searches using that choice and displays student info using a tabular form
* Has a retrial option

1. void edit()

* Has a retrial option
* Uses ID number to search the data to be edited

1. void deletion()

* Searches the wanted student using student ID and deletes the entry

1. inline void retrial(int a)

* Gives the choice to go back to the main menu or not
* If not displays “have a good day” message
* If invalid choice it will ask if the user would like to try again

1. Void saveinfo()

* Writes a student’s info to a text file called “user.txt”.
* Info including name, age. Sex, regeistration date, payments date, student ID and total pay.

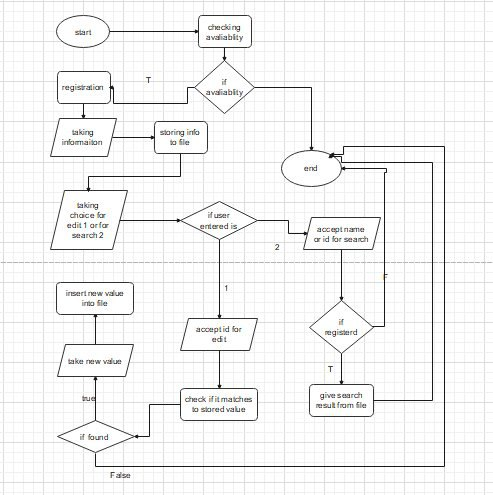
1. Void getinfo()

* Gets student info from different text files
* Pulls info about the courses taken from "Course count.txt"
* Pulls info about the students ID from "ID.txt"
* Pulls info about the students count from "Student count.txt"

1. void menu()

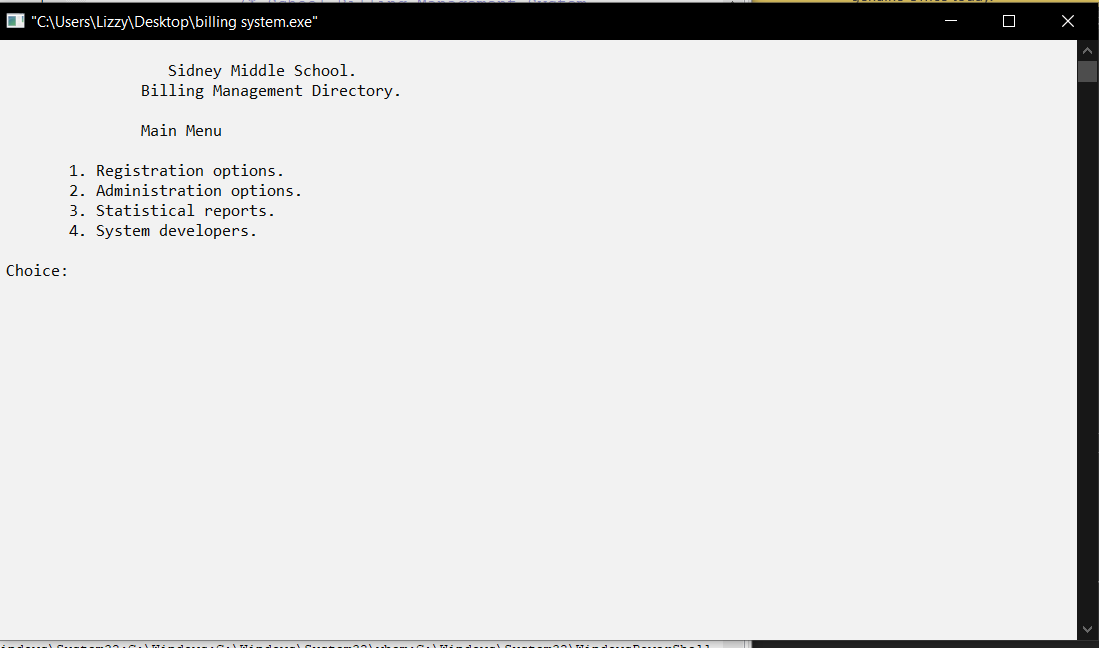
* This is the master table/ menu which displays list of choice for the user to choose from
* It calls 3 functions to their respective cases and define the developers option by listing out the group members name and their instructor respectively

3.3 Flow Chart



3.4 Demonstration pictures from the system

Main menu



The main screen or the first page is the first thing a user will see when he/she runs the program. And it has these options to choose from

1. Registration Options
2. Administration options
3. Statistical reports
4. System developers

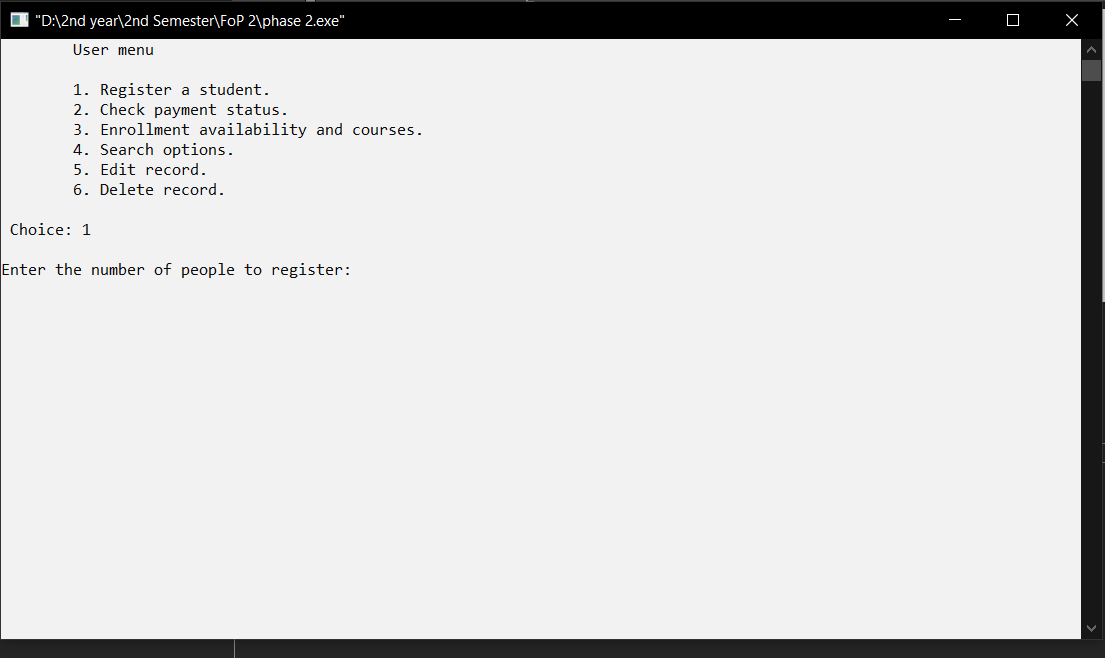
When the user selects option 1 or the registration option the following will appear

User menu

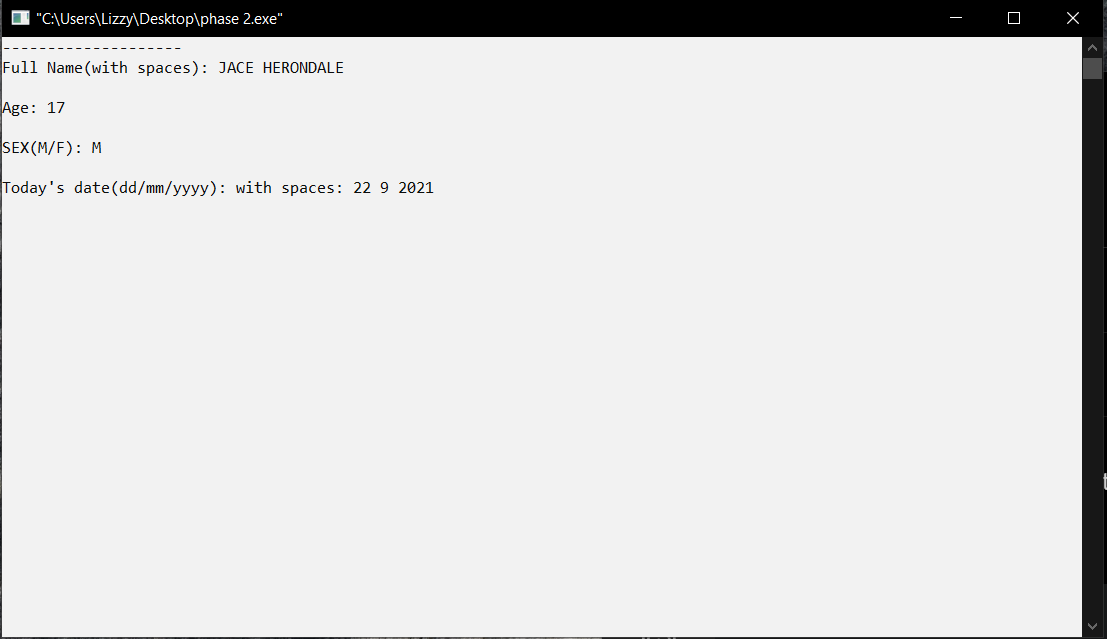


The user menu has 6 options Registration, Payment status check, Enrollment availability, search option, edit and finally delete. The code will display the user menu and will wait for our response to choose form the list.

User menu



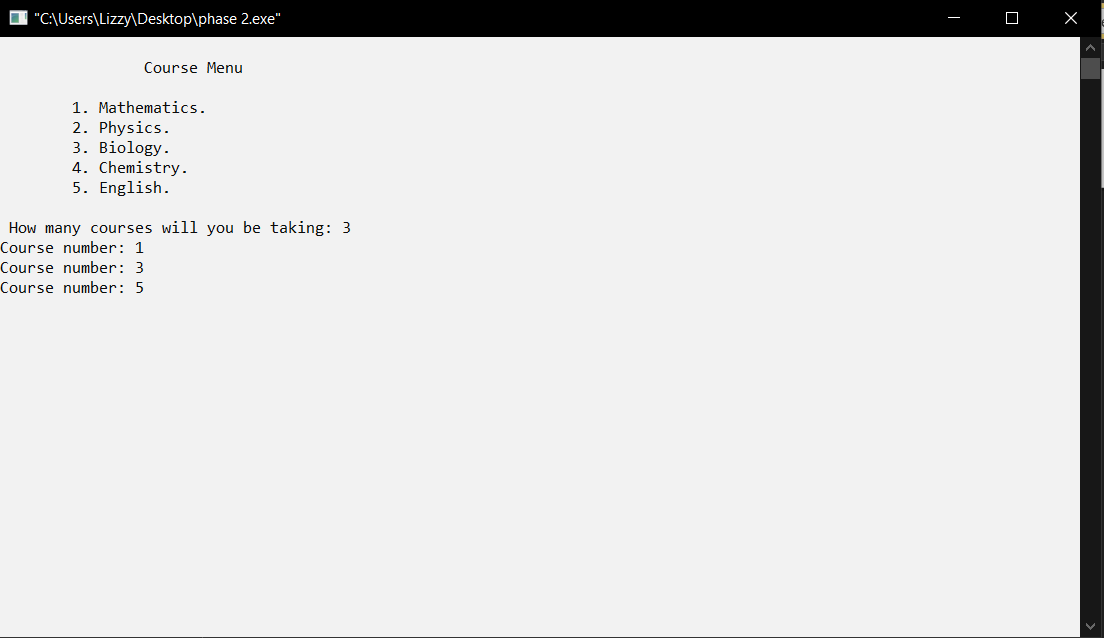
When we enter 1 which is “Register a student” option the code asks for the number of students to register. After we input number of people for example lets say we want to register 1 person the it will ask the user to enter full name, age, sex and date respectively and it looks like this.



After entering the student’s full information the system will proceed to the next process which is asking the user to input the number of courses the student will be taking. For example 3 courses.

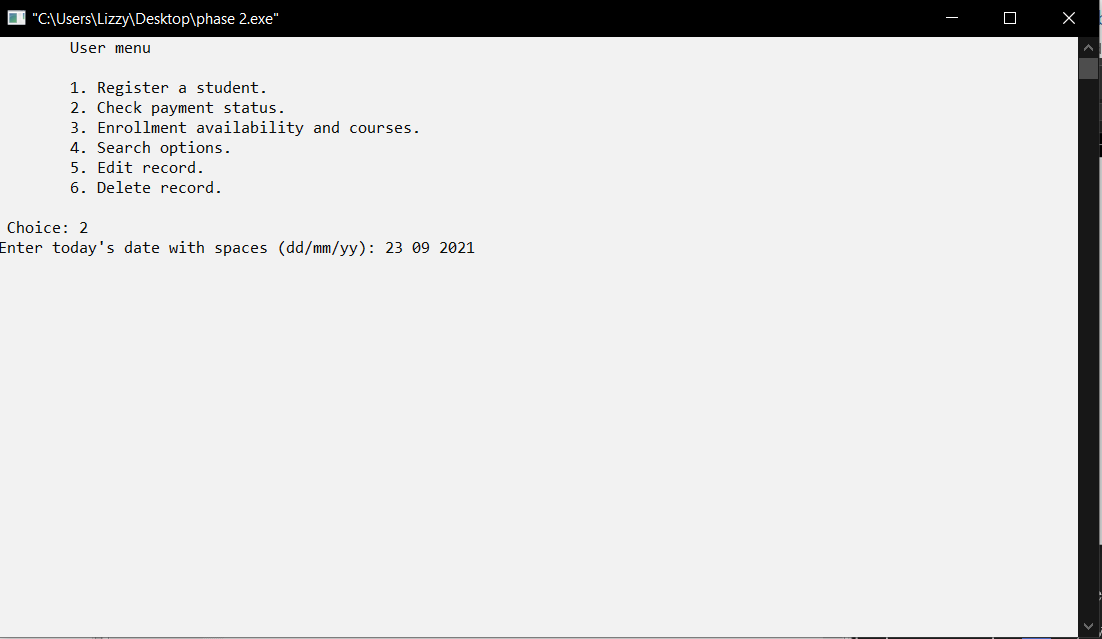


The system then asks for the course number from the course list. For this demonstration we chose Mathematics, Biology and English.

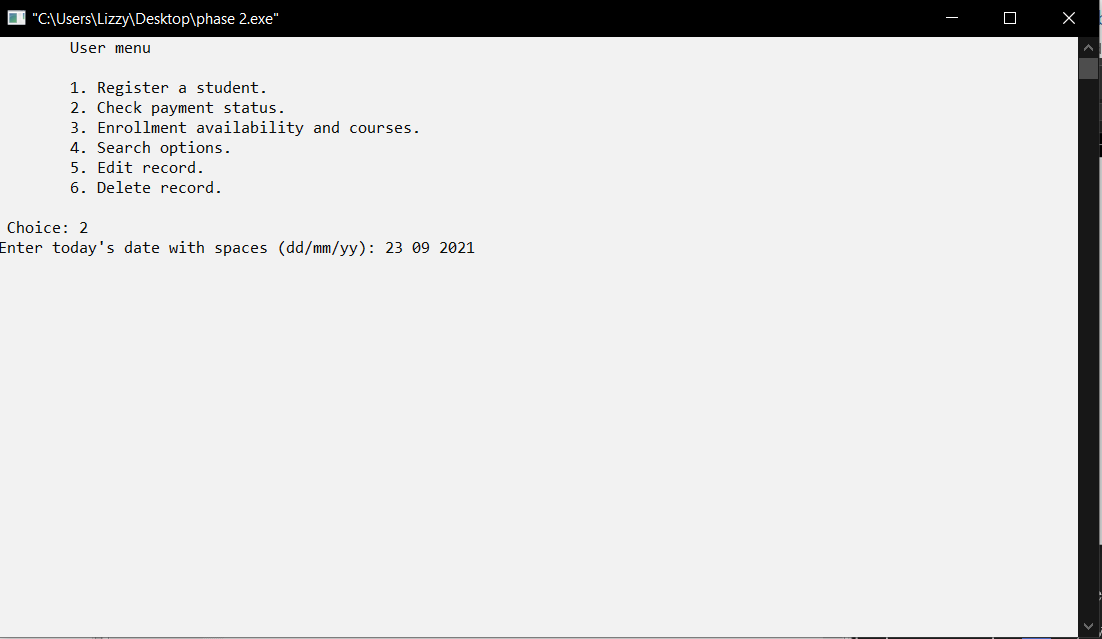


Payment option is offered and that will be the end of registration. The system provides a receipt calculating the total payment from the individual course fee. So the receipt will contain name, age, sex, payment date courses that will be take and the total payment.

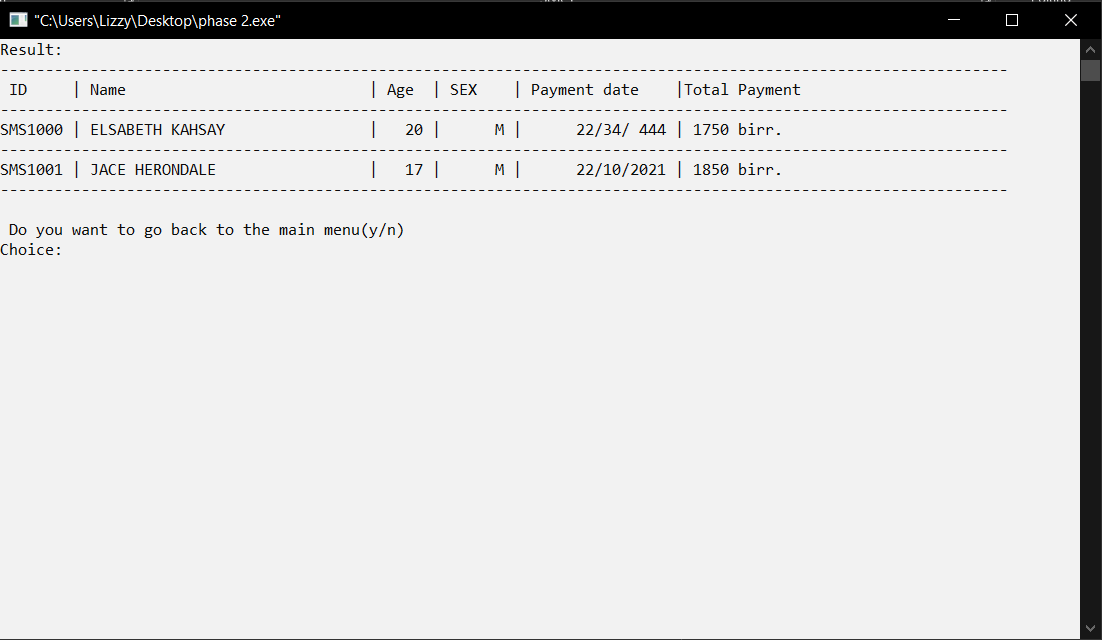




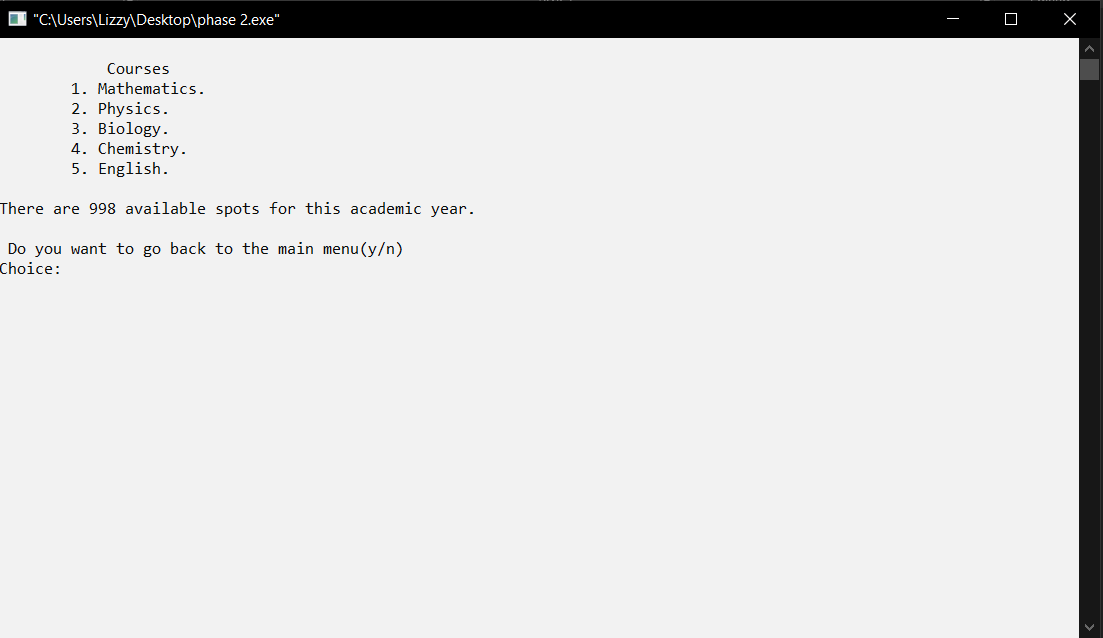
Back to the user menu when we select 2, to check payment status the system asks today’s day and after we enter the date the following will be displayed



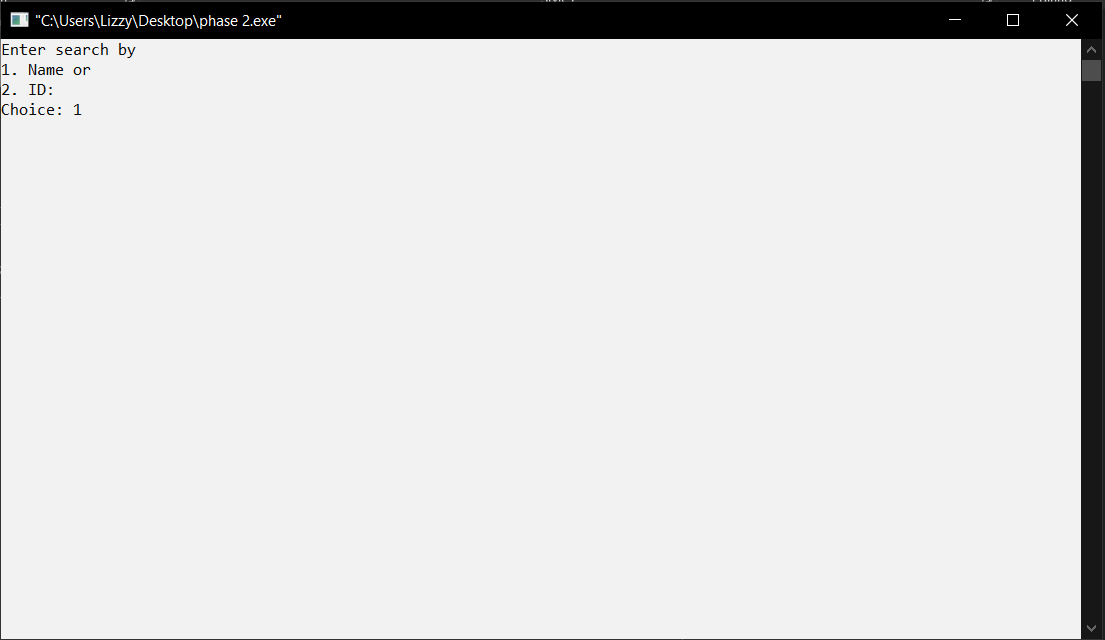
The system displays the students payment status



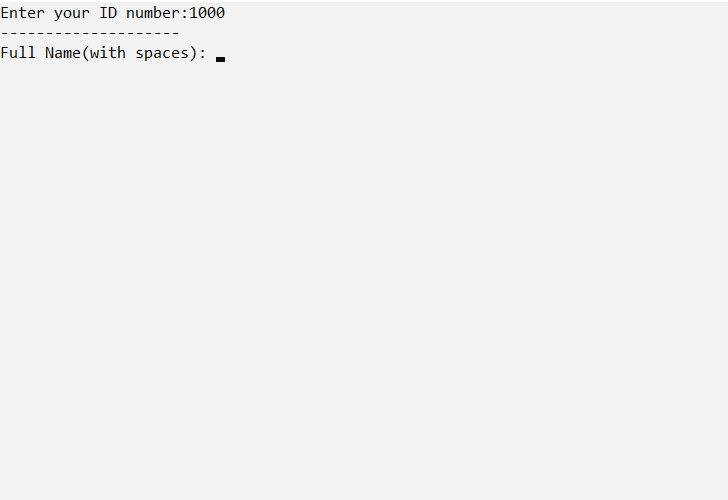
The third option which is enrollment availablity and courses the system displays list of courses available with the number of available spots or openings.

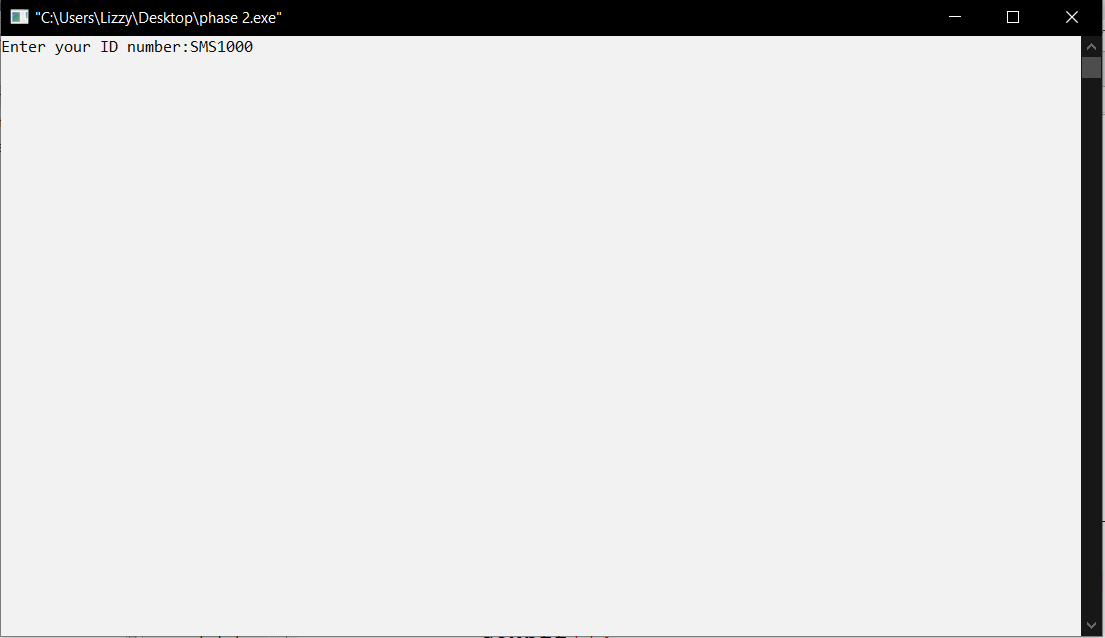


The fourth option which is the search option. It gives the user to search for the student using either name or ID for demonstration we will see search by name.

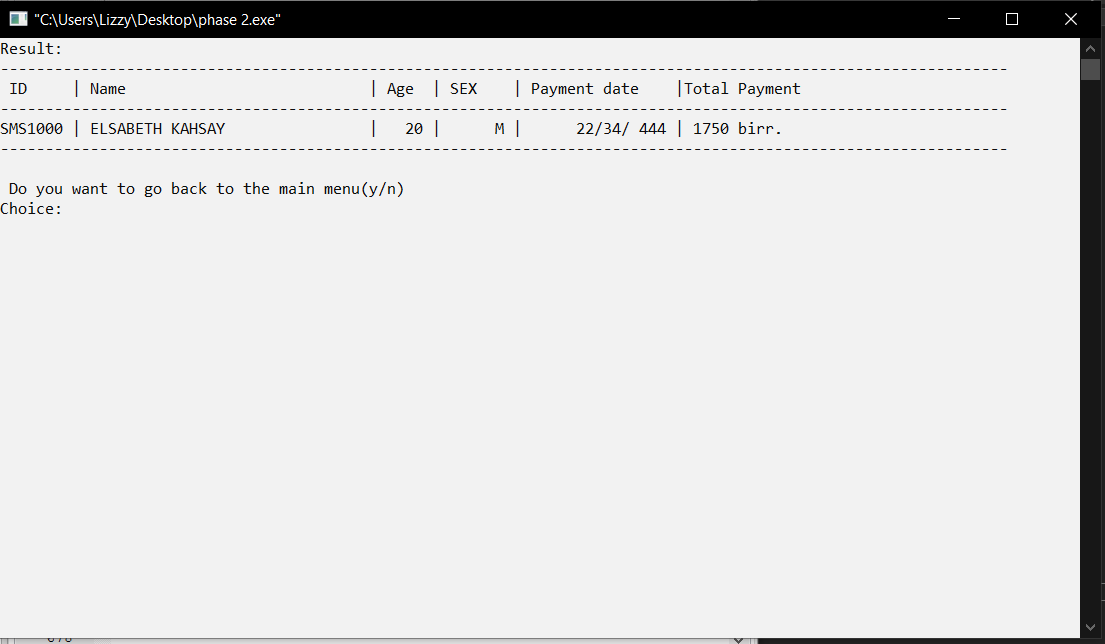


The fifth option is Edit option.

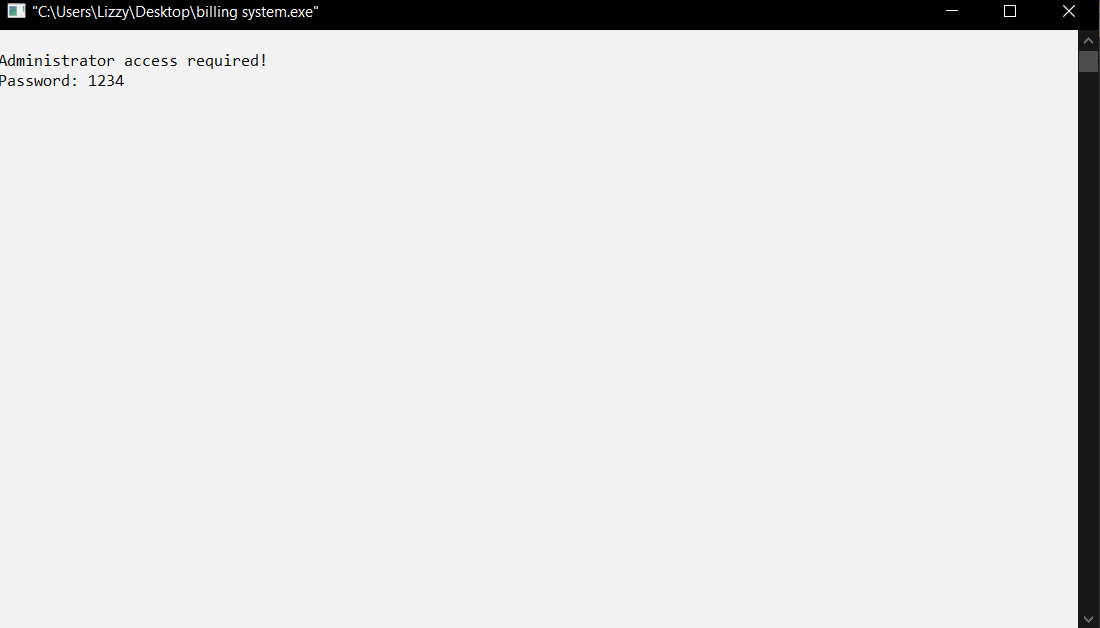


The last option which is the delete option deletes entery from the system. It asks for ID number of the specific student’s ID.

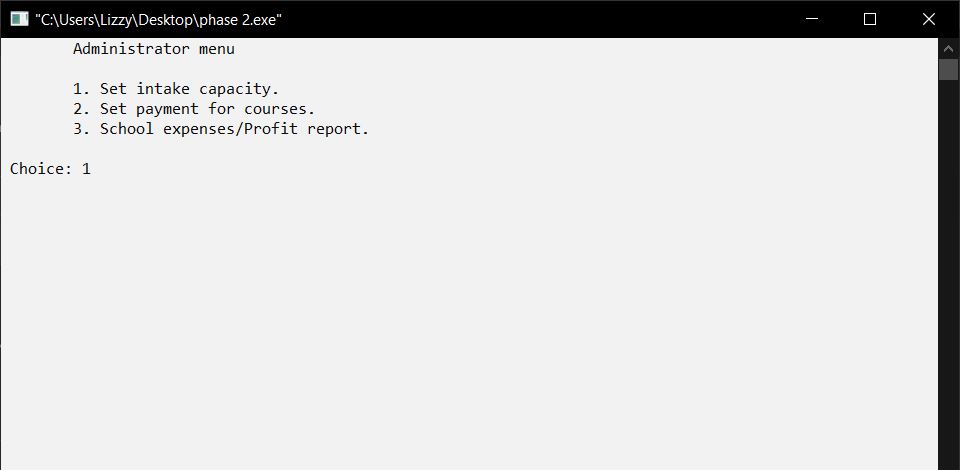
Previously we had 2 students info but now its just one.



When option 2 Administration options is selected



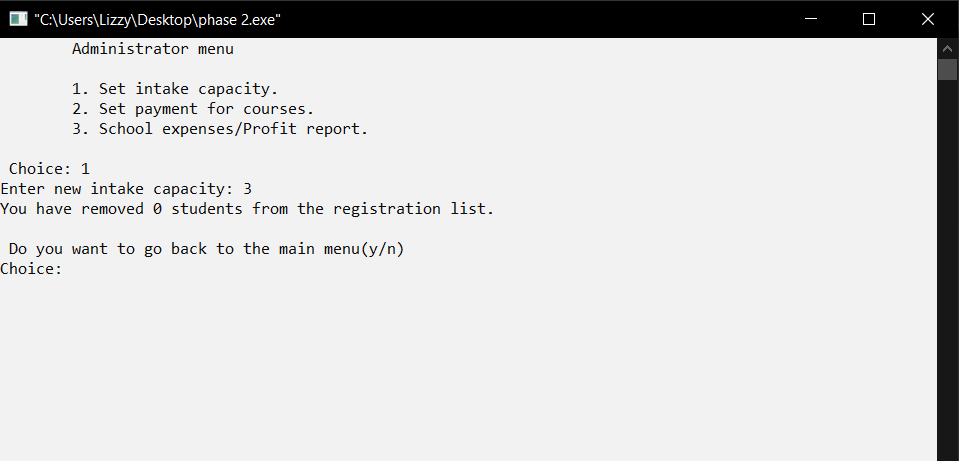
After choosing the admin option it asks for password for security. So the user should provide the correct password to be granted access to alter or edit information in the system.



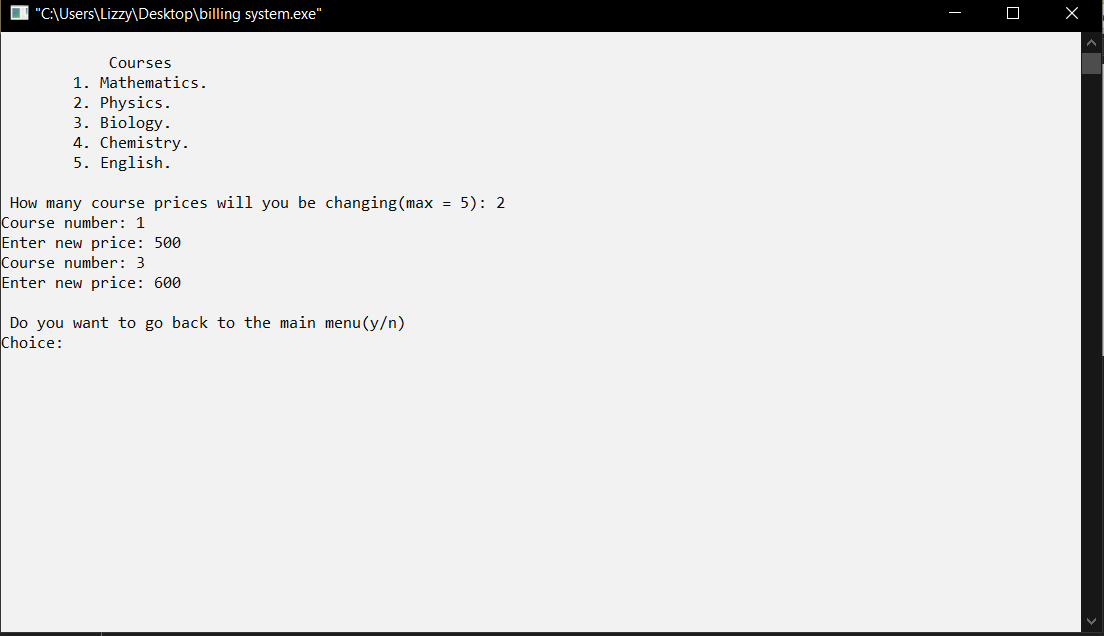
After being granted access to the system as an admin the system lists 3 Administration menu namely

1. Set intake capacity
2. Set payment for courses
3. School expenses/profit

When we select the first option the system will procced to the following

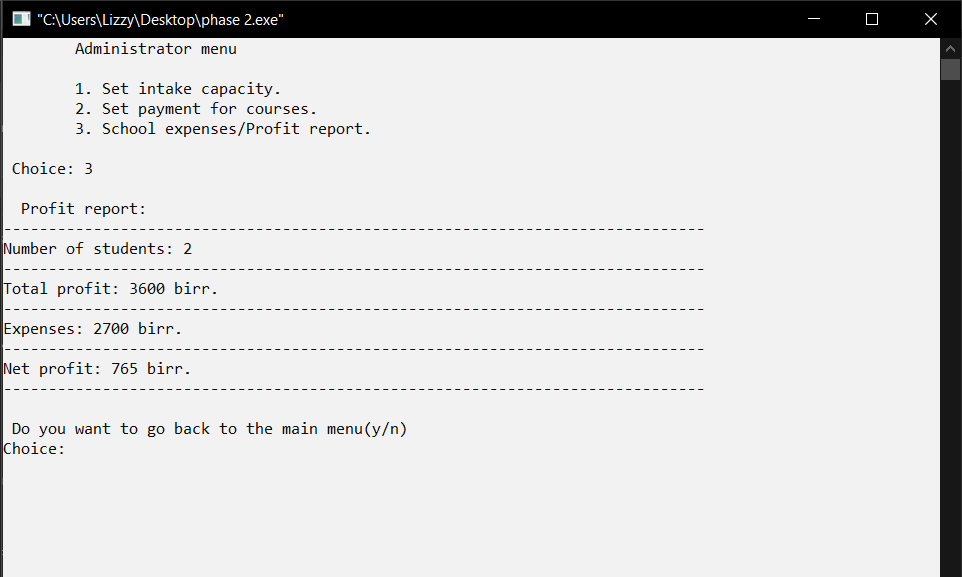


When we choose the second option or to set payment for a course th system will ask for the number of courses to be changed to edited.

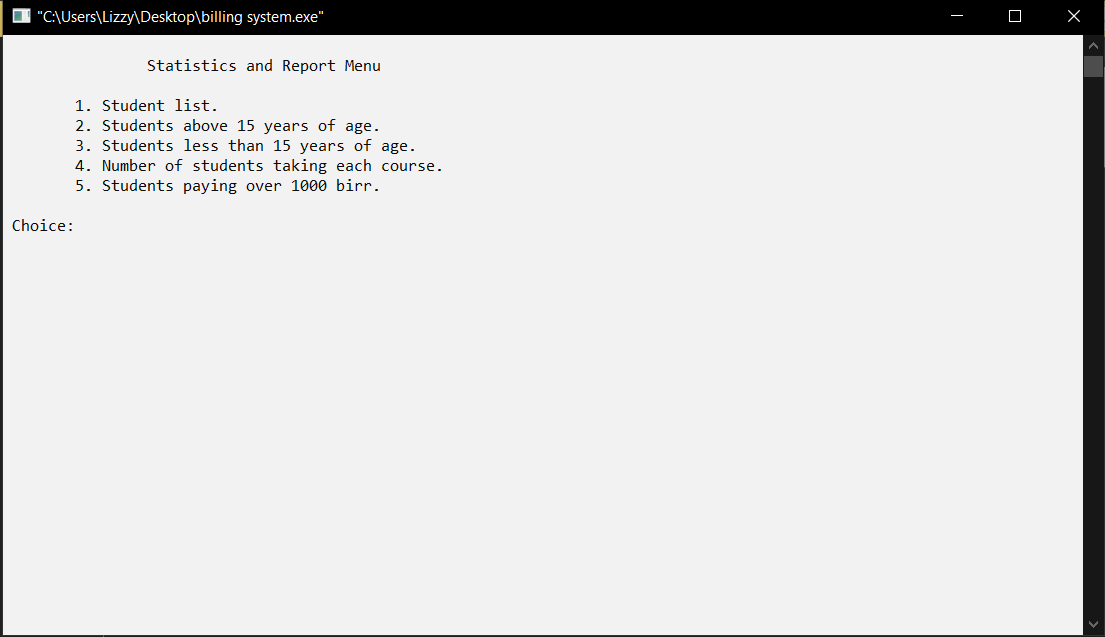


After providing the number of courses to be altered/ changed the system will again ask the user/admin to enter the number which identifies the course and to enter the new price.

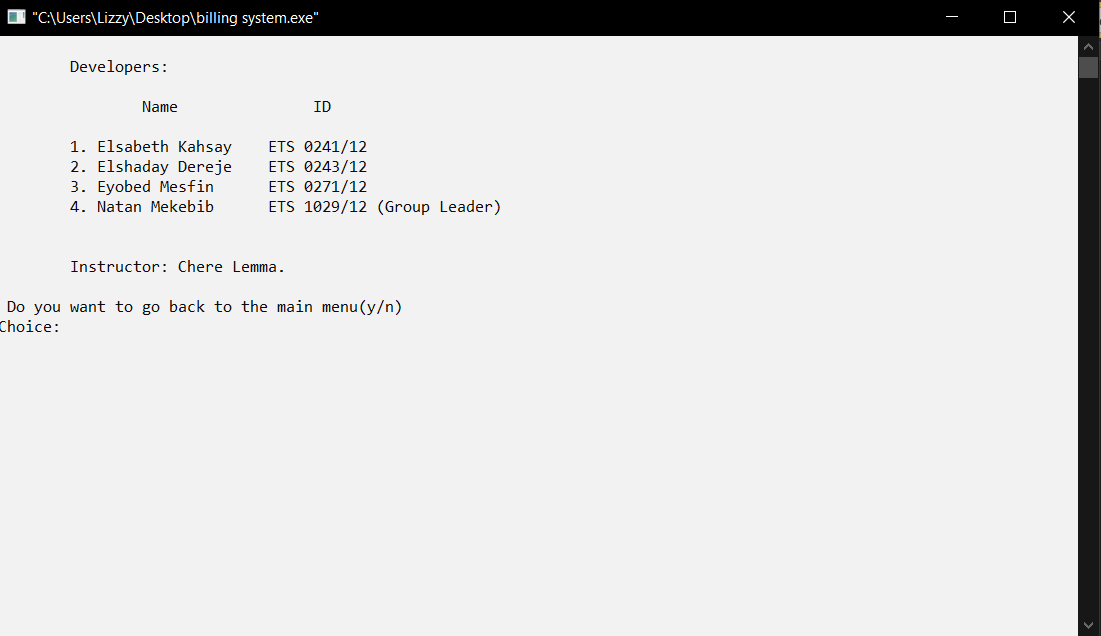
When we select the third option which is to check school expenses or profit the following will be displayed given that we registered 2 students with their full information. It calculates the total profit, expenses and net profit using the algorithm which is in the code.



When option 3 stastical reports option is selected from the main menu the following will be displayed and it will present reports per requested.



Finally When option 4 System developers option is selected the system will display the list of developers and their instructor respectively with the option to go back to the main menu.



1. **Conclusion**

We made this school billing management system is made by using C++ programming language as a project. It includes many concepts like structures, functions, file management, vectors and many more things that we’ve learnt in this semester. And by using all those we’ve made this system which is very simple to use, user friendly and very useful.

This project has helped us implement all the things we learnt in class which was a good learning opportunity.

1. **Reference**

*Chapter 5 pointers (I, II)*

*Chapter 7 (I, II) user defined data types*

*Chapter 8 file management*

*Chapter 9 templates*

*W3schools*

*Geeksforgeeks.org*