جامعة عفت

EFFAT UNIVERSITY

CS 1131

Advanced Programming - Fall 2021

Author: Layal Ghryani S20106409 Razan Almahdi S20106649

Instructor: **Akila Sarirete**

Date Last Edited: December 21, 2021

Contents

1 Overview

This program is designed to implement several new classes representing a school, and its students and courses. These classes will show the dynamic of the students, courses, and the arrays that contain them in an interactive form.

2 Introduction

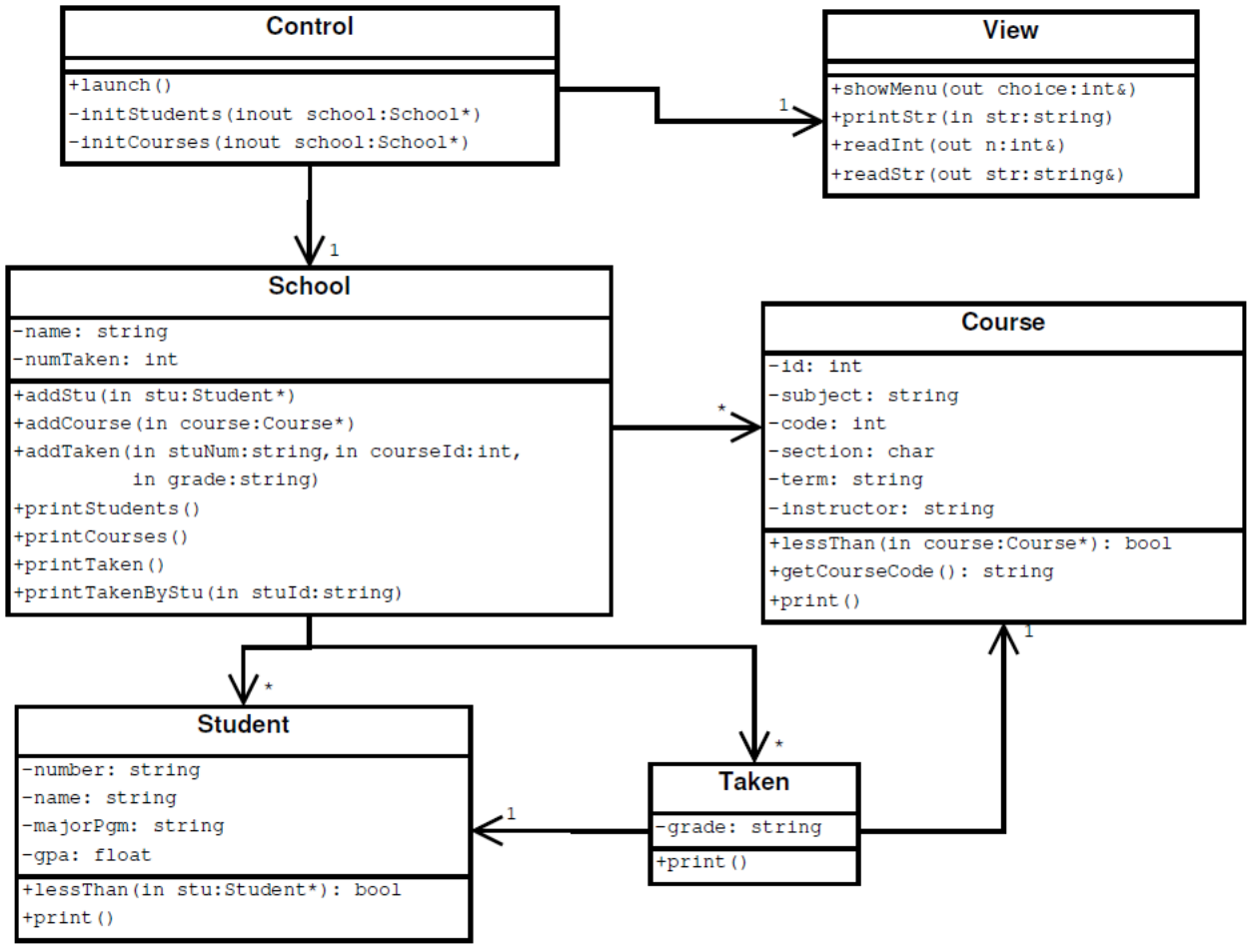
2.1 Goal and Learning outcomes

For this project, you will write a C++ program to manage the data for a school with students and courses. You will implement your program using objects from the different classes, based on a UML class diagram provided for you.

- practice implementing a design that is given as a UML class diagram
- implement a program separated into control, view, entity, and collection objects
- work with statically allocated and dynamically allocated arrays

3 Problem Statement and Design

3.1 UML class diagram



User (Layal and Razan)

3.2 Implement the student class

The Student class is the class where the constructs for all student information is stored. it includes the constructor, destructor, print, getter and setter functions, and a comparing boolean function that organizes the students in the array according to their names.

3.3 Implement the Course class

The Course class contains all the constructs of all courses. It has a constructor, desctructor, print, getter and setter function, And comparing function that organizes the courses according to their subject, then code, then section.

3.4 Implement the Taken class

The taken class that refers to both course and student class. It groups up both of them by using getter functions, and constructing them into a structure of its own for the taken class.

3.5 Implement the DynArray class

The DynArray class relates to the student class. It is used to create a dynamic array, create student objects and add them, find student objects within the array, and eventually print them.

3.6 Implement the StatArray class

The StatArray class relates to the course class. It is used to create a dynamic array, create course objects and add them, find course objects within the array, and eventually print them.

3.7 Implement the School class

The School class relates to Course, Student, and Taken classes. It enables us to add student objects, course objects, and taken objects to the dynamic arrays, and then print them by specification according to the option chosen in the View + Control class.

3.8 Implement the Control class

The Control class is the backbone of the program. It contains all of the student and course objects, and the information for each object, and launches the entire program. It also has a switch function that checks the input imported from the View class, and based on it, it will connect to the School class functions.

3.9 Implement the View class

The View class is a class with no private members. Its purpose is to output the menu of options, and prompt the user for input. Then, it passes the input on to the Control class to activate the switch, and so on.

4 Executions

Case 1: Makefile creation of executables

```
C:\>cd C:\Users\user\Downloads\project

C:\Users\user\Downloads\project>ls
Control.cc          DynArray.cc          School.cc           Student.cc          View.cc
Control.h           DynArray.h           School.h            Student.h           View.h
Course.cc           Makefile             StatArray.cc        Taken.cc
Course.h            ProjectTestDriver.cc StatArray.h         Taken.h

C:\Users\user\Downloads\project>make project
g++ -c ProjectTestDriver.cc
g++ -c Student.cc
g++ -c Course.cc
g++ -c Taken.cc
g++ -c DynArray.cc
g++ -c StatArray.cc
g++ -c School.cc
g++ -c Control.cc
g++ -c View.cc
g++ -o project ProjectTestDriver.o Student.o Course.o Taken.o DynArray.o StatArray.o School.o Control.o View.o
```

Case 2: the menu of the program and a list of all courses and students

```
--------------------------------------------------------------------------------
--------------Hello and welcome to Effat University's database!-----------------
--------Here is a brief list of all the students and courses in our database:



------------------------------------------------------
--------------------The Students----------------------
------------------------------------------------------
-- Student:  Razan Almahdi
-- Student:  Aicha Sidiya
-- Student:  Aida Alulu
-- Student:  Layal Ghryani
-- Student:  Leen Almalki
-- Student:  Mariam Kadi
-- Student:  Nour Mohammed
-- Student:  Lujain Salam
------------------------------------------------------
--------------------The Courses-----------------------
------------------------------------------------------
-- Course:  CS
-- Course:  CS
-- Course:  CS
-- Course:  CS
-- Course:  MATH
-- Course:  MATH
-- Course:  CS
-- Course:  GANT
-- Course:  GENG
-- Course:  GISL
-- Course:  GARB
-- Course:  BIO
-- Course:  GFREN
-- Course:  GGERM
-- Course:  STAT
-- Course:  STAT
-- Course:  ANIM
-- Course:  ARCH
-- Course:  ARCH
------------------------------------------------------
------------------------------------------------------
```

Case 3: printing all of the student objects and their information in detail

```
What would you like to do?
  (1) Print students
  (2) Print courses
  (3) Print all courses taken
  (4) Print courses taken by student
  (5) Add course taken by student
  (0) Exit

Enter your selection:
1

Effat University -||Students:

Student:  S20177632  Aicha Sidiya      CS               GPA:  3.75
--------------------------------------------------------------------
Student:  S21574411  Aida Alulu      ARCH             GPA:  3.51
--------------------------------------------------------------------
Student:  S20640553  Layal Ghryani      CS               GPA:  3.75
--------------------------------------------------------------------
Student:  S11034592  Leen Almalki      ARCH             GPA:  3.55
--------------------------------------------------------------------
Student:  S50686199  Lujain Salam      CA               GPA:  3.45
--------------------------------------------------------------------
Student:  S20213597  Mariam Kadi      MKT              GPA:  3.64
--------------------------------------------------------------------
Student:  S34955605  Nour Mohammed      ECE              GPA:  3.60
--------------------------------------------------------------------
Student:  S20106649  Razan Almahdi      CS               GPA:  3.75
--------------------------------------------------------------------
```

Case 4: print printing all of the courses objects and their information in detail

```
What would you like to do?
  (1) Print students
  (2) Print courses
  (3) Print all courses taken
  (4) Print courses taken by student
  (5) Add course taken by student
  (0) Exit

Enter your selection: 2
```

```
Effat University -||Courses:

Course ID: 71102    ANIM   1405   Section: 3  Term: F21     Instructor: MG
---------------------------------------------------------------------------
Course ID: 65019    ARCH   1030   Section: 1  Term: S22     Instructor: HL
---------------------------------------------------------------------------
Course ID: 69810    ARCH   2043   Section: 2  Term: F22     Instructor: HL
---------------------------------------------------------------------------
Course ID: 10032    BIO   112   Section: 1  Term: F21     Instructor: SS
---------------------------------------------------------------------------
Course ID: 13494    CS   1001   Section: 1  Term: S20     Instructor: HK
---------------------------------------------------------------------------
Course ID: 43929    CS   1131   Section: 3  Term: F21     Instructor: AS
---------------------------------------------------------------------------
Course ID: 84932    CS   2021   Section: 2  Term: F21     Instructor: HK
---------------------------------------------------------------------------
Course ID: 64913    CS   2111   Section: 3  Term: F20     Instructor: ZB
---------------------------------------------------------------------------
Course ID: 24492    CS   3067   Section: 1  Term: S22     Instructor: KK
---------------------------------------------------------------------------
Course ID: 58764    GANT   143   Section: 2  Term: F21     Instructor: MK
---------------------------------------------------------------------------
Course ID: 43209    GARB   161   Section: 2  Term: W20     Instructor: AD
---------------------------------------------------------------------------
Course ID: 32009    GENG   162   Section: 1  Term: F20     Instructor: DA
---------------------------------------------------------------------------
Course ID: 33280    GFREN   140   Section: 3  Term: S22     Instructor: CA
---------------------------------------------------------------------------
Course ID: 60431    GGERM   180   Section: 2  Term: F21     Instructor: FJ
---------------------------------------------------------------------------
Course ID: 45871    GISL   171   Section: 3  Term: S20     Instructor: MK
---------------------------------------------------------------------------
Course ID: 78431    MATH   101   Section: 1  Term: F21     Instructor: TB
---------------------------------------------------------------------------
Course ID: 92184    MATH   201   Section: 2  Term: S22     Instructor: MM
---------------------------------------------------------------------------
Course ID: 44937    STAT   101   Section: 2  Term: F20     Instructor: OK
---------------------------------------------------------------------------
Course ID: 80992    STAT   201   Section: 1  Term: S22     Instructor: OK
---------------------------------------------------------------------------
```

Case 5: print all the taken objects [it returns this message if there are no taken objects]

```
What would you like to do?
  (1) Print students
  (2) Print courses
  (3) Print all courses taken
  (4) Print courses taken by student
  (5) Add course taken by student
  (0) Exit

Enter your selection: 3

------------------------------------------------------------
Effat University -||Taken: None!!!
------------------------------------------------------------
```

Case 6: check for courses taken by particular student [it returns this if the ID given does not relate to a student or a student who has not taken any courses]

```
What would you like to do?
  (1) Print students
  (2) Print courses
  (3) Print all courses taken
  (4) Print courses taken by student
  (5) Add course taken by student
  (0) Exit

Enter your selection: 4
Enter your Student ID: S20106146

-------------------------------------------------
----------No courses taken by this student!-------
-------------------------------------------------
```

Case 7: adding a student object into the taken array [it returns this if the student ID or course ID is incorrect]

```
What would you like to do?
  (1) Print students
  (2) Print courses
  (3) Print all courses taken
  (4) Print courses taken by student
  (5) Add course taken by student
  (0) Exit

Enter your selection: 5
Enter your Student ID: S20106146
Enter your Course ID: 1
Enter your Grade: A+

-----------------------------------------------
Student ID/ Course ID was not found. Try again!
-----------------------------------------------
```

Case 8: adding a student object into the taken array [it returns this if the student and course ID are both correct]

```
What would you like to do?
  (1) Print students
  (2) Print courses
  (3) Print all courses taken
  (4) Print courses taken by student
  (5) Add course taken by student
  (0) Exit

Enter your selection: 5
Enter your Student ID: S20106649
Enter your Course ID: 80992
Enter your Grade: A+
--Student:  0x6000011b4d20

------------------------------------------------
Razan Almahdi has been successfully registered in this course.
------------------------------------------------
```

Case 9: printing the taken class [this time, the taken class has an object or more and is not empty]

```
What would you like to do?
  (1) Print students
  (2) Print courses
  (3) Print all courses taken
  (4) Print courses taken by student
  (5) Add course taken by student
  (0) Exit

Enter your selection: 3

------------------------------------------------------------
Effat University --Taken:
Taken:  Razan Almahdi STAT201 A+
------------------------------------------------------------
```

Case 10: check courses taken by a particular student [it returns this if the given ID is correct and the student has taken courses]

```
What would you like to do?
  (1) Print students
  (2) Print courses
  (3) Print all courses taken
  (4) Print courses taken by student
  (5) Add course taken by student
  (0) Exit

Enter your selection: 4
Enter your Student ID: S20106649

Effat University Taken By  Razan Almahdi
Course ID: 80992    STAT   201   Section: 1  Term: S22     Instructor: OK
--------------------------------------------------------------------------------
```

Case 11: using the exit option to terminate the program. the arrays will be deleted to prevent memory leakage.

```
deleting Taken Array . . .

-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:
-|| Course:


deleting static array . . .


-|| Student:  Aicha Sidiya
-|| Student:  Aida Alulu
-|| Student:  Layal Ghryani
-|| Student:  Leen Almalki
-|| Student:  Lujain Salam
-|| Student:  Mariam Kadi
-|| Student:  Nour Mohammed
-|| Student:  Razan Almahdi


deleting dynamic array . . .
```

5 Conclusion

This 1 month project was a roller coaster filled with tears and happiness. Teamwork was assessed extremely as we had to put both of our patience and knowledge and understanding skills to the test. It was fun to an extent. We really struggled and in the end we achieved what we wanted and the program works. To Future programmers even though this took longer time to create we learn and discovered new skills that we didn't know about and We personally thought we wouldn't reach the end of this creation journey and here we are. Thank you so much Dr.Akila for this eye opening project!

5.1 Teamwork distribution

This project was extremely time consuming for 2 people, we faced a problem in the beginning but we made it work. We divided the work equally 50/50. Even when one of us struggled in their class implementation we still worked together and finished. It was hard doing this project during a period we had other submissions and quizzes, but we found time and did it. We cannot lie about the fact that even when splitting the task it was a big risk because we had to put both of our knowledge and understanding and patience, and we encountered lots of errors of all sorts, yet we did it together and successfully. Below is the functional dependencies table for both members:

Task	Member
student class	Razan
course class	Layal
school class	Razan
dynarray class	Razan
statarray class	Layal
taken class	Razan + Layal
control class	Razan + Layal
report	Razan
presentation	Layal