Sultan Qaboos University
Department of Computer Science
COMP2102: Problem Solving and Programming, Fall 2011
Assignment #1 – Due Date: Saturday 15th October 2011

The purpose of this assignment is to get you acquire skills with program design and development and functional decomposition using functions to implement your solution. Read the assignment problem, assignment parts, submission policy and grade distribution sections.

Problem

An input data file **students.txt** (See Figure1) stores students' registrations for last semester. Develop a solution that reads and processes unknown number of students' data from the input file and generates a grade report document (an output file named **report.txt**) that shows for every student, the student id, student name, courses registered, total credits attempted, total credits earned, total grade points earned, the grade point average (gpa), and the number of students processed. See Figure 3 for the format of the output file.

Use the following calculation rules:

- Total credits attempted include credits of all courses registered except credits of courses with **I** or **W** grades.
- Total credits earned include credits attempted except credits of courses with **F** grade.
- Total grade points earned equals the summation of (credit *numeric grade points) for all courses attempted except the courses with **P** grade. Use the following numeric grade points:

A	A-	B+	B	B-	C+	C	C-	D+	D	F
4.0	3.7	3.3	3.0	2.7	2.3	2.0	1.7	1.3	1.0	0.0

- Grade Point Average (GPA) = (Total grade points earned)/(total credits of courses attempted except courses with **P** grade). GPA calculations for the student 11111 are illustrated below. Note that grades **P**, **I**, **W** are not included in the GPA calculations.

Course	Credit	Grade	Numeric Grade Points	Value of Grade
COMP2102	3	A	4.0	12.0
PHYS2102	3	B	3.0	9.0
LANC1001	3	C+	2.3	6.9
ARAB1100	3	F	0.0	0.0
COMP1200	2	I		
CROP4543	2	P		

Total credits attempted=14
Total credits earned=11
Total grade points earned=27.9
Grade Point Average=27.9/12=2.325=2.33

Part1: Problem Solving:

Use the program development process discussed in chapter 2 to develop your solution to the given problem, and submit a word document containing the following and name your document "**A1_XXXXX.doc**" (where XXXXX is your university id):

a. Step1**: Problem Definition**: Define the given problem in a concise and clear statement of your own words.

b. Step2: **Requirement Specification:** Determine input/output data, its sources, named constants needed and processing steps.

c. Step3: **Design**: Using modularization approach, provide a full algorithm of your solution (in pseudocode). Verify your algorithm (i.e. *desk checking*) with the hand example illustrated for the student 11111 above. Also consider file error situations.

d. Step4: **Implementation**: Submit this step as your source C++ program (.cpp) for Part2.

e. Step5: **Testing and Verification**: Submit the output produced by your program for each student, and for file error situations.

Part2: Programming: Perform the following tasks:

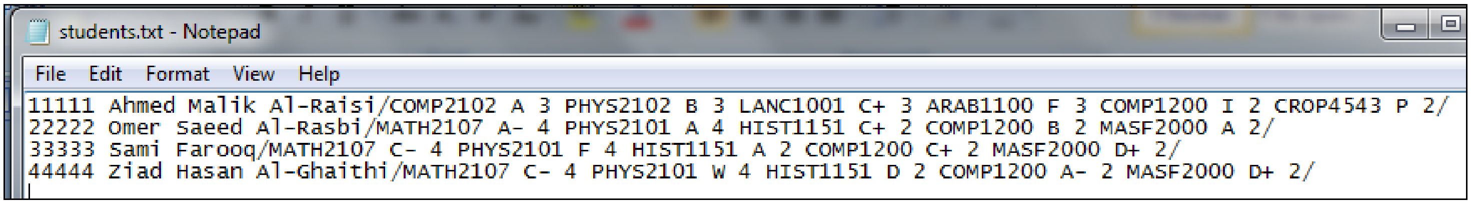
1. Create the input data file **students.txt** shown in Figure1. Each line contains student id, student name, a forward slash, followed by a list of student courses data including a course id, grade scored and course credit hours. A forward slash is used to end the student courses list.

2. Implement your algorithm as a C++ program(named **A1_XXXXX**, where XXXXX is your id number) using a separate function for each of the following tasks:

Function	***Prototype***
Display the report header as shown in the output file showing your name and university id (see Figure 3)	*void header(ofstream&);*
Read a single student id and name. Hint: use getline(…) to read the student name.	*void ReadIDAndName(ifstream&, int&,string&);*
Read a single course details: course id, grade, credit. Return true if the end of courses list is reached otherwise return false.	*bool ReadCourseData(ifstream&,string&,string&,int&);*
Get a grade numeric points. Called by Calculate(…)	*double GetGradeNumericPoints(string);*
Calculate running total credits attempted, total credits earned,	*void Calculate(string,int, int&, int&,double&, int&);*

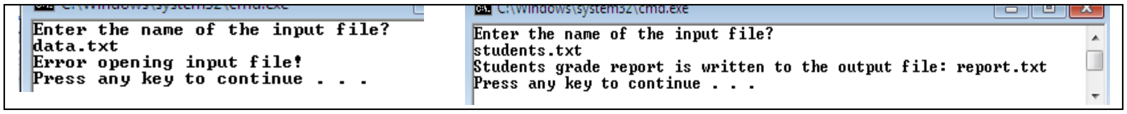
total grade points, and total credits used based on received course grade and credits.	
Function main acts as a director opening and closing files, calling various functions, aggregating results and writing to output file. It should terminate program if there is error opening files.	

3. Declare output file name “reports.txt” and grades numeric points (e.g. A 4.0) as global constants.
4. Read the input file name from the user.
5. Format the output of your program as shown in Figures 2 & 3.



```
11111 Ahmed Malik Al-Raisi/COMP2102 A 3 PHYS2102 B 3 LANC1001 C+ 3 ARAB1100 F 3 COMP1200 I 2 CROP4543 P 2/
22222 Omer Saeed Al-Rasbi/MATH2107 A- 4 PHYS2101 A 4 HIST1151 C+ 2 COMP1200 B 2 MASF2000 A 2/
33333 Sami Farooq/MATH2107 C- 4 PHYS2101 F 4 HIST1151 A 2 COMP1200 C+ 2 MASF2000 D+ 2/
44444 Ziad Hasan Al-Ghaithi/MATH2107 C- 4 PHYS2101 W 4 HIST1151 D 2 COMP1200 A- 2 MASF2000 D+ 2/
```

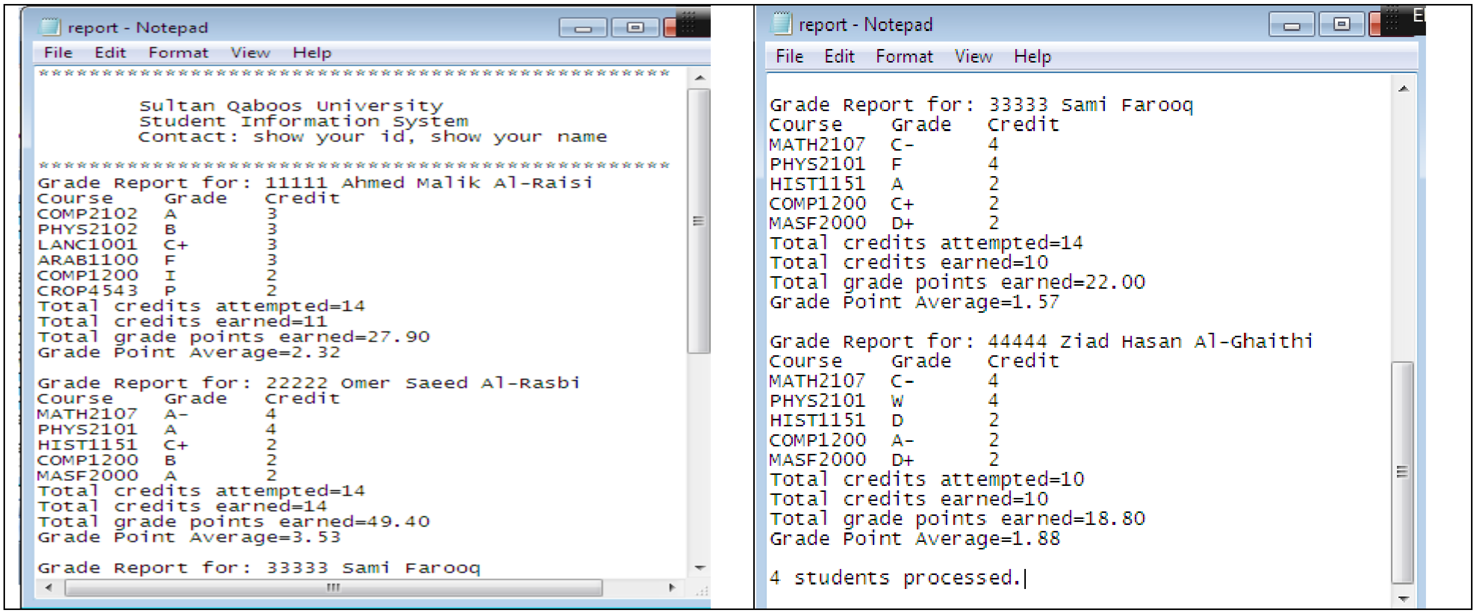
Figure 1: sample input data file “students.txt”



```
Enter the name of the input file?
data.txt
Error opening input file!
Press any key to continue . . .
```

```
Enter the name of the input file?
students.txt
Students grade report is written to the output file: report.txt
Press any key to continue . . .
```

Figure 2: Sample Screen Outputs



```
**************************************************


         Sultan Qaboos University
         Student Information System
         Contact: show your id, show your name

**************************************************
Grade Report for: 11111 Ahmed Malik Al-Raisi
Course    Grade   Credit
COMP2102  A       3
PHYS2102  B       3
LANC1001  C+      3
ARAB1100  F       3
COMP1200  I       2
CROP4543  P       2
Total credits attempted=14
Total credits earned=11
Total grade points earned=27.90
Grade Point Average=2.32

Grade Report for: 22222 Omer Saeed Al-Rasbi
Course    Grade   Credit
MATH2107  A-      4
PHYS2101  A       4
HIST1151  C+      2
COMP1200  B       2
MASF2000  A       2
Total credits attempted=14
Total credits earned=14
Total grade points earned=49.40
Grade Point Average=3.53

Grade Report for: 33333 Sami Farooq
```

```

Grade Report for: 33333 Sami Farooq
Course    Grade   Credit
MATH2107  C-      4
PHYS2101  F       4
HIST1151  A       2
COMP1200  C+      2
MASF2000  D+      2
Total credits attempted=14
Total credits earned=10
Total grade points earned=22.00
Grade Point Average=1.57

Grade Report for: 44444 Ziad Hasan Al-Ghaithi
Course    Grade   Credit
MATH2107  C-      4
PHYS2101  W       4
HIST1151  D       2
COMP1200  A-      2
MASF2000  D+      2
Total credits attempted=10
Total credits earned=10
Total grade points earned=18.80
Grade Point Average=1.88

4 students processed.
```

Figure 3: Sample output file format

Submission Policy and Grade Distribution:

1. Upload the two parts of your solution (part1: problem solving document **A1_XXXXX.doc** and part2: C++ source file **A1_XXXXX.cpp**, where XXXXX is your university id) to Assignment 1 link in your Moodle class.
2. Late submission & copying policies will be applied as described in the course outline.
3. Break down of the points will be as follows:

a.	Proper document naming and submission	/1
b.	Part1 (Problem Solving)	/5
c.	Comments and indentation	/2
d.	Use of named constants	/2
e.	Files declaration, opening and closing	/4
f.	Appropriate use of functions	/10
g.	Correct function main logic & processing	/7
h.	Correct output & results	/5
i.	Program free of compilation errors & warnings	/2
j.	Program free of run-time errors	/2
k.	**Total**	**/40**