# _LITfinalLOGO

# SUMMER EXAMINATIONS 2012

**Tuesday, 15th May 2012, 9.30 a.m. – 11.30 a.m.**

**KSDEM\_8\_Y1**

**Course:** Bachelor of Science (Hons) in Software Development

**Year:** One

**Subject:** Programming

**Time Allowed:** 2Hours

**Instructions: 1.** You **MUST** answer **Question ONE (1)**

Answer **ANY OTHER TWO (2)** questions.

**2.** Marks for **Question ONE (1)** are **40 marks**.

All other questions are **30 marks.**

**3.** Start each question on a new page.

**4.** Write the question number at the top of each page.

**5.** Circle the numbers of the questions you answer at the front of your answer book.

**Additional Attachments Exam Materials to accompany this paper:**

### A. None

**Internal Examiners: External Examiners:**

Tom Costello Mr. Paul Powell

**Q. 1**

1. What would be the value of z displayed by each of the following code fragments
   1. **int x = 4, y = 3, z;**

**z = x - y \* x;**

**cout << z << endl;**

**(4 marks)**

1. **int x = 5, z;**

**double y = 12.5;**

**z = y / x;**

**cout << z << endl;**

**(4 marks)**

1. **int x = 10, y = 6, z;**

**x + = 2;**

**z = x % y;**

**cout << z << endl; (4 marks)**

1. Re-write each of the following code fragments correcting all syntax errors
   1. **int difference;**

**if { x ! = y }**

**difference = { x – y };**

**cout >> “Values differ by ” >> difference >> endl;**

**else**

**cout >> “Zero Difference\n”;**

**(4 marks)**

**Q.1 contd.**

1. **int x;**

**cin >> x;**

**switch if x**

**[**

**case 1**

**cout << “Option 1\n”;**

**break;**

**case 2**

**cout << “Option 2\n”;**

**break;**

**otherwise**

**cout << “Invalid option\n”;**

**]**

**(4 marks)**

1. **cout << “Displaying even numbers \n”;**

**for ( int x = 0, x < 10, x++ )**

**{**

**if x % 2 == 0**

**cout << x << endl;**

**}**

**cout << “Displaying odd numbers\n”;**

**int y = 0;**

**do while y < 10**

**{**

**if y % 2 != 0**

**cout << y << endl;**

**y++;**

**}**

**(4 marks)**

**Q.1 contd.**

1. What values will be displayed when each of the following code fragments is executed
   1. **for (int x = 24; x > 10; x - = 4 )**

**{**

**cout << x << endl;**

**}**

**(4 marks)**

* 1. **for ( int row = 0; row < 3; row++ )**

**for ( int col = 0; col < 2; col++)**

**cout << row << endl;**

**(4 marks)**

* 1. **int x = 36;**

**while ( x > 0 )**

**{**

**cout << x << endl;**

**x /= 3;**

**}**

**(4 marks)**

* 1. **for ( int x = 1; x < 15; x += 3 )**

**{**

**cout << x << endl;**

**if ( x % 5 = = 0 )**

**break;**

**}**

**(4 marks)**

**(Total 40 Marks)**

**Q. 2** The monthly life insurance premium charged by an insurance company is dependent on the age of the applicant. A basic premium of €50 is charged and an additional number of €12.50 increments is added based on the applicants current age as per the following table

|  |  |
| --- | --- |
| **Age** | **Number of increments** |
| less than 30 | 0 |
| 30 to 49 | 1 |
| 50 to 59 | 3 |
| 60 to 64 | 6 |

Applicants aged 65 and over will not be considered.

Write a program that

1. will allow the user to enter the current date and the date of birth of an applicant (in the form dd mm yyyy) and will compute the applicants age

**(10 marks)**

1. If the applicant is ineligible due to age, the program should terminate with an appropriate message displayed.

**(4 marks)**

1. where applicable, will based on the applicants age ascertain the number of increments that apply to this applicant

**(10 marks)**

1. will calculate and display the required monthly premium

**(6 marks)**

**(Total 30 Marks)**

**Q. 3** A store supplies 8 different spare parts for a machine, the parts having

part numbers from 101 to 108. The price of the parts in € are given in the

following table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Part Number** | 101 | 102 | 103 | 104 | 105 | 106 | 107 | 108 |
| **Price** | 2.44 | 13.50 | 11.20 | 8.50 | 2.33 | 16.40 | 82.00 | 1.25 |

A customer’s order consists of a list of 8 pairs of integers where each pair consists of a part number and the quantity required. Some of the quantity values may be zero. Such an order is contained in a text file called “Order.txt”.

Write a program that will

1. initialize a 1-D array with the prices from the above table.

**(5 marks)**

1. read the contents of the file “Orders.txt” into a 2-D array.

**(10 marks)**

1. display an invoice for the order as per the following sample

**Part Number Quantity Unit Price Cost**

**102 1 13.50 13.50**

**105 4 2.33 9.32**

**Total Cost 22.82**

**(15 marks)**

**(Total 30 Marks)**

**Q. 4** A **class CStudent** is added to a project for an application called **ResultsApp** that is to be used to manage the exam results for a group of students .

A student will be awarded marks for three individual subjects, namely Chemistry, Biochemistry and Project. To pass the exam a student must obtain a minimum of 40 marks in each subject. If a student has passed the exam he will be awarded a grade as per the following table

|  |  |
| --- | --- |
| **Average Mark** | **Grade String** |
| >= 70 | H1 |
| 60 to < 70 | H2.1 |
| 50 to < 60 | H2.2 |
| 40 to < 50 | Pass |

The code below shows a definition of the class.

**class CStudent**

**{**

**private:**

**string m\_Name;**

**int m\_Chemistry;**

**int m\_Biochemistry;**

**int m\_Project;**

**public:**

**CStudent (void);**

**CStudent (string name);**

**CStudent (string name, int chem, int biochem, int proj);**

**void SetChemistry(int mark);**

**void SetBiochemistry(int mark);**

**void SetProject(int mark);**

**string GetName(void);**

**int GetChemistry(void);**

**int GetBiochemistry(void);**

**int GetProject(void);**

**bool HasName(string name);**

**bool IsAPass(void);**

**string GetGrade(void);**

**};**

Write definitions for all the member functions of the class

**a)** The first constructor

**CStudent (void)**

should initialize the name data member to an empty string and the

other data members to zero

**(4 marks)**

1. The second constructor

**CStudent (string name)**

should initialize the appropriate data member to the value passed as an argument and the other data members to zero

**(4 marks)**

1. The third constructor

**CStudent (string name, int chem, int biochem, int proj)**

should initialize the appropriate data members to the values passed as arguments

**(4 marks)**

1. The **Set** functions should insert the mark for the particular subject

**(3 x 1 marks)**

1. The **Get** functions should return the appropriate data member

**(4 x 1 marks)**

1. The function **HasName** checksfor a matching name in the object

**(2 marks)**

1. The function **IsAPass** should return true only if all of the subject marks are greater than or equal to 40

**(3 marks)**

1. The function **GetGrade** will return the appropriate grade string if the student has passed all exams If student has failed it will return an empty string

**(6 marks)**

**(Total 30 Marks)**

**Q. 5** The application **ResultsApp** mentioned in **Q.4** is designed for a group of up to 40 students and will instantiate a global collection of **class CStudent** objects and a global variable to contain the actual number of students as per the following code

**CStudent list[40];**

**int numStudents = 0;**

The global data will be made persistent by reading from and writing to a file each time the application is opened and closed.

Each time a new student is added to the system the **numStudents** variable will be incremented.

The application will display a menu driven interface that will allow the user to access this global collection. Menu options and associated callback functions will allow a user to :

1. initialize the group of students by entering the name of each student. This should cause a **class CStudent** object to be instantiated for each student and for the object to be inserted in the collection.

The callback function has a prototype **void DoInitializeGroup(void).**

Write a definition for this function.

**(10 marks)**

1. select a subject and then insert the marks for each student in the group.

The callback function has a prototype **void DoEnterSubjectMarks(void)**

Write a definition for this function.

**(10 marks)**

1. display a list of the names and grades of all the students who have passed the exam.

The callback function has a prototype **void DoDisplayPassList(void)**

Write a definition for this function.

**(10 marks)**

**(Total 30 Marks)**