National University of Computer and Emerging Sciences

FAST School of Computing   Spring 2020   Islamabad Campus

CS-217: Object Oriented Programming

Remote Final Exam

Attempt Time: 3 Hours
Submission (on Google Classroom and through email) Time: 15 minutes

Monday, 6th July, 2020

Total Marks: 180

Course Instructors

Mr. Muhammad H. Mustafa, Dr. Naveed Ahmad, Dr. Kifayat Alizai, Dr. Muhammad Aleem, Mr. Jawad Hasan, Mr. Hassan Raza

Instructions:

1. The final exam will be attempted offline in the student's own handwriting (in a readable way).
2. The students will use A4 size blank white sheets to attempt the exam (portrait format unless a diagram or table requires landscape).
3. Each sheet of the A4 size paper **MUST have the Roll Number, Name, the course code, name of the course and Signature** of the student at the top.
4. If you are reading these instructions, you should **NOT** spread the word that the students who would draw a **BIG** smiley on the first page of their answer sheet will get 5 bonus marks. If you know how relative grading works, you will not want anyone else to draw the smiley. Keep reading the remaining instructions, they are important.
5. Students will use cam-scanner, MS lens, or an equivalent application to scan and convert their hand-written answer sheets into a **SINGLE** pdf file (keeping the correct order of pages and question numbers), which they will submit on **Google Classroom** and **MUST also** email to the email address (of the concerned course instructor) within 15 minutes after the 3 hours attempt time. Submissions after 30 minutes may not be accepted. **Try to submit soon after 3 hours of attempt time and do not wait for 15 minutes to be elapsed.** Students should carry a clean scanning that is free from any marks/stains etc.
6. All students must use the standard **file name** format (Full course code - Roll number e.g. **CS-217-19i-0123**). The subject of your **email** must also follow the same format **(CS-217-19i-0123)**.
7. For a proven cheating/ plagiarism case, the student will get an F grade even if the student had opted for S/U grade, and the case will be referred to DDC (Department's Disciplinary Committee). Instructors will conduct viva of randomly selected students, or in case of doubt (significantly different attempt as compared to the past performance in the course or matching attempt with other students). Plagiarism includes sharing an attempt to other students (copy providing). Students who are not able to satisfactorily answer instructor's questions (based on the exam as well as slightly lateral but related concepts) during viva will also be considered as plagiarism cases.

	Q-1	Q-2	Q-3	Q-4	Q-5	Q-6	Total
Marks	30	30	30	30	30	30	180

Question 1

Part-A [15 Marks]

i. Write a function **allocateMemory** in C++ which takes a pointer variable (think about its multiplicity of indirection) of type integer and 6 integer arguments. Allocate dynamic memory of 6D array to that pointer. Size of array is passed in arguments.

```
// allocating memory for a 6D array of size 3x5x7x9x8x4.
allocateMemory(ptr, 3,5,7,9,8,4);
```

ii. Write another function **fillArray** which takes arguments as above function with addition of an extra argument of initial value. Fill the values in such a way that it starts with initial value and fills the other indexes in series.

```
// filling the array with numbers starting from 20.
fillArray(ptr, 3,5,7,9,8,4, 20);
```

iii. Also write another function **deallocateMemory** for the deallocation of above allocated memory. You must decide arguments of this function.

iv. Is it necessary to pass pointer variable by reference in above three functions or not?? Justify your answer in two to three sentences.

Part-B [15 Marks]

Write a recursive function **oddFivesInOctal** which takes a number (Decimal) as argument and returns whether octal representation of that number has odd number of digit 5 or not?
You are not allowed to use if or if-else statement outside this function.

```
bool oddFivesInOctal(int n){
//your code
}
```

oddFivesInOctal(45) should return false because octal representation of 45 is 55 and there **are two occurrences of digit 5** in 55. Two is not odd.
oddFivesInOctal(46) should return true because octal representation of 46 is 56 and there is **one digit 5** in 56. One is odd.
oddFivesInOctal(2605) should return true because octal representation of 2605 is 5055 and there are **three occurrences of digit 5** in 5055. Three is odd.

Question 2 [30 Marks]

Create a SavingsAccount class to store data of savers (account holders). Your class should match the following specifications.

1. Each instance of the class contains a private data member savingsBalance indicating the amount the saver currently has on deposit, saver’s name, saver’s CNIC, account number (this has to be unique) and a member to store saver status (if savingsBalance > 10000 then status changes to gold otherwise silver).

2. Class also has a data member annualInterestRate to store the annual interest rate which will be the same for all the savers.

3. Implement a default constructor with minimum savingsBalance of 100, and other fields as well. Also create a parameterized constructor if a customer wants to open an account with a different starting balance.

4. Provide member function calculateMonthlyInterest that calculates the monthly interest by multiplying the savingsBalance by annualInterestRate divided by 12; this interest should be added to savingsBalance.

5. Provide a recursive member function turnInToGold that returns the number of months a saver need to become gold member

6. Provide a member function modifyInterestRate that sets the annualInterestRate to a new value.

7. Provide a function that returns the total number of account holders with the bank.

8. Write main() to test class SavingsAccount . Instantiate two different objects of class SavingsAccount, s1 and s2, with balances of $2000.00 and $3000.00, respectively. Set the annualInterestRate to 3 percent. Then calculate the monthly interest and print the new balances for each of the savers. Then set the annualInterestRate to 4 percent, calculate the next Month’s interest and print the new balances for each of the savers. Also print the months it will take for both to become gold savers of bank.

9. Write code in three separate files as shown below.

//SavingsAccount.h//

```
class SavingsAccount{

}
```

//SavingsAccount.cpp//

//definition of functions

//main.cpp//

//main to show how different functions are called

Question 3 [30 Marks]

In architectural drawings, the distances are measured in feet and inches according to the English system of measurement. There are 12 inches in a foot. The length of a living room, for example, might be given as 15’–8”, meaning 15 feet plus 8 inches. The hyphen isn’t a negative sign; it merely separates the feet from the inches. Figure 1 shows typical length measurements in the English system. Suppose you want to create a drawing or architectural program that uses the English system, it will be convenient to store distances as two numbers, representing feet and inches. Develop a complete class with proper constructor and destructor functions as well as set and get functions to model and program the Distance representation in the English System. Also include two functions, getDistance to get input from the user and showDistance to display the distance in the prescribed format. We need you to keep track of how many instances of the class are created in main or any other function. The class should also provide the following overloaded operator capabilities:

1. Overload the addition operator (+) to add two Distances.

2. Overload the less than operator (<) to compare two Distances does

3. Overload the addition assignment operator (+=).

4. Overload the increment operator (++)

5. Overload the stream insertion operator (<<) for displaying distance.

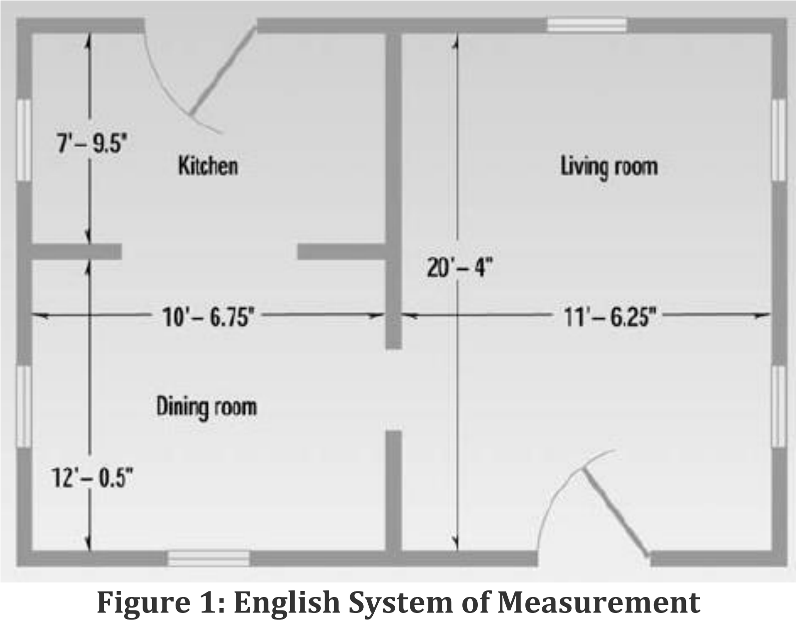


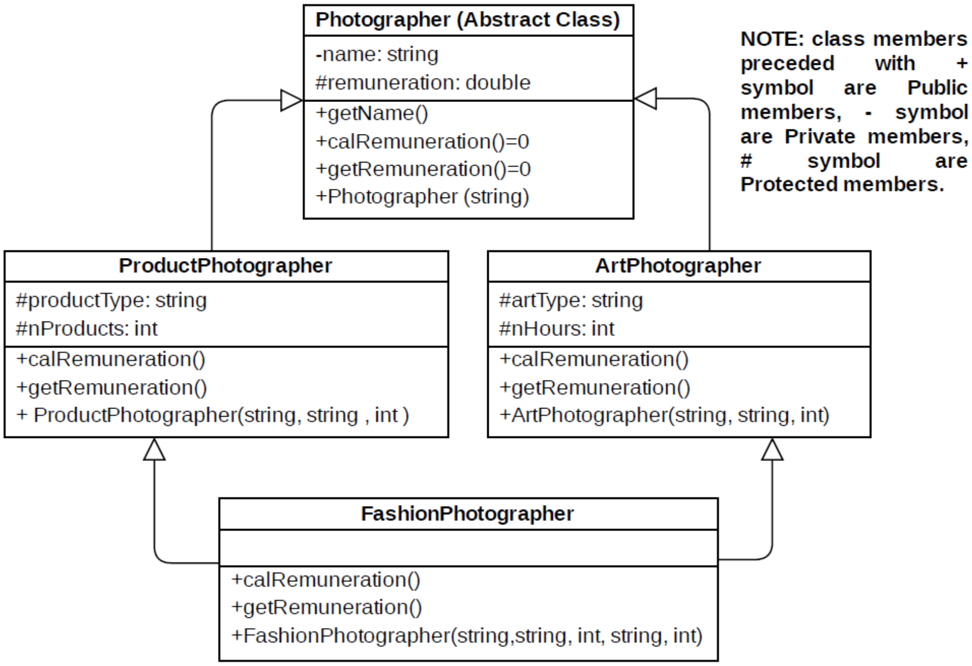
Figure 1: English System of Measurement

Scoring Criteria	
Feature	**Marks**
1. Appropriate Declaration/definition of class	8 marks
2. Correct Syntax	4 marks
3. Logic of overloaded operators	15 marks
4. Comments or Explanation of code	3 marks

Question 4 [30 Marks]

Consider a Photo studio company that provides professional photography services. To target the skills required in different domains, the company hires several kinds of photographers such as 1) *Product Photographer*, 2) *Art Photographer*, and 3) *Fashion Photographer*.

To design and implements this system, implement 3 Classes (as shown in below Figure) which represent the specific photographer type as mentioned above, and the 4th Class (named "Photographer") an abstract class representing a collection of common attributes and function related to all the other 3 concrete classes.



NOTE: class members preceded with + symbol are Public members, - symbol are Private members, # symbol are Protected members.

The **Photographer** class should contain ***name*** and ***remuneration*** (a double type value for storing service charges). Moreover, the functions provided by the Photographer class are ***getName*** (returns name of the photographer), ***calRemuneration*** (calculates the remuneration and stores into ***remuneration*** data-item), ***getRemuneration*** (returns the remuneration value), and a parameterized constructor.

The **ProductPhotographer** class (a kind of Photographer) contains the attributes ***productType*** (name of a product) and ***nProducts*** (the number of products for which the photography services was done). Moreover, the class provides its own implementation of ***calRemuneration*** and ***getRemuneration*** functions and a parameterized constructor. The remuneration of a product photographer is calculated using 3000 Rs per product.

The **ArtPhotographer** class (a kind of Photographer) contains attributes ***artType*** (name of the art type) and ***nHours*** (number of hours the photography services was provided). Moreover, the class provides its own implementation of ***calRemuneration*** and ***getRemuneration*** functions***,*** and a parameterized constructor. The art photographer charges for his services at the rate of 1800 Rs per hour.

The ***FashionPhotographer*** class (a kind of ProductPhotographer and ArtPhotographer) provides its own implementation of ***calRemuneration*** and ***getRemuneration*** functions***,*** and a parameterized constructor. The fashion photographer charges for his services using 1000 Rs per hour plus 2000 Rs per product.

In the main function, create a product photographer, an art photographer, and a fashion photographer. The product photographer Mr **Usman** photographed 7 different furniture items. Mr **Ahmed** (the art photographer) photo shot for performing arts for 8 hours. The fashion photographer Mr **Daud** photographed a fashion wear brand for 4 hours covering 15 different cloth wear items. The main function should use a dynamic binding concept to invoke ***calRemuneration, getRemuneration,*** and ***getName*** functions for these 3 types of photographers separately. The typical program output should be as follows:

```
 Product Photographer USMAN earned: 21000
---------------------------------------------
 Art Photographer AHMED earned: 14400
---------------------------------------------
 Fashion Photographer DAUD earned: 34000
```

Question 5 [30 Marks]

1000			1001			1002		
A	B	C	A	B	C	A	B	C
1,2,3,4,5	20,22, 23,24 25,26	30,31,32 ,33,34, 35	1,2,9,10, 11,12	27,28,29	36,37,38	13,14,15 ,16	20,21	39,40

In any data processing and analytics system, data is normally collected from data source(s), then fed into a repository (a database or a data structure) and finally an analysis is performed on the data. Let us imagine that we have received the data and now we want to store it for querying and analysis. **Strictly and only using STL (Standard Template Library) of C++,** your job is to

1. Write complete code to produce a data structure as shown in the above diagram**. (10 Marks)**

2. Once the data structure is created, then populate it with the data shown in the figure. **(5 Marks)**

3. Next, write code to perform following analytics on this data structure
   i. Combine all the values having same Letter value/label **(5 Marks)**
      a. Example: all the entries with label 'A' are 1,2,3,4,5,9,10,11,12,13,14,15
      b. Example: all the entries with label 'B' are 20,22,23,24,25,26, 27, 28, 29
      c. Example: all the entries with label 'C' are 30,31,32,33,34,35,36,37,38,39,40

   ii. Formulate a question/query from your own by combining 3 different algorithms together to get the answer for your proposed question/query. **(10 Marks)**

Notes for getting higher marks:

1. The code which is elegant and efficient in terms of processing. Code written in bits and pieces where different actions are independently being performed is not considered as an efficient and elegant code.
2. The analytics which are unique and interesting. For this purpose, may be, you can combine different algorithms to produce a new one for diverse analytics
3. Overall unique code for the entire problem

Question 6 [30 Marks]

MailMerge is an interesting utility of Microsoft Office suite. One of its key usage is in sending personalized emails to a list of recipients by linking two text files - one that contains comma-separated personalized data including email id, and the other that contains the email text with some placeholders. In the remainder of the question, I will call the files as **"data.txt"** and **"body.txt"** respectively.

The process of sending emails is simple. MailMerge reads the email text from *body.txt* file. This text contains some placeholders, which MailMerge replaces with the relevant information from *data.txt* file. A placeholder is the field number from *data.txt* and is enclosed in the double angular brackets. Like, <<1>> represents that the data in the first field from *data.txt* file should be replaced in the email text here, and <<2>> means that the data from the second field should be replaced here, and so on. You can see that the fields are numbered from 1 to N, where N is the total number of fields in a record in *data.txt* file. Moreover, *data.txt* file contains one comma-separated record on each line and there may be M number of records in the file at a time, where M can be any positive number. Here, for simplicity, you can assume that the first field of every record will always contain the email id of the recipient.

Write a C++ function, naming MailMerge, to read the two files (*data.txt* and *body.txt*) and generate corresponding emails. Instead of sending actual emails (as we haven't studied any email-sending library), you shall create a new file for each email and write the output email text in it. You shall name the output file as the email id (the first field in *data.txt* file).

Scoring Criteria	
Feature	**Marks**
1. Correctly reading data.txt and body.txt	5+5
2. Correct and efficient logic to generate output text	10
3. Correctly writing the output files	5
4. Correct C++ syntax and commenting	5

The following table shows a sample scenario for your understanding.

Sample Input and Output	
File Name	**Contents**
Sample body.txt file	Dear <<2>>, Find below your report of the mock final exam with the relevant observations: Q1: <<3>> / 10 marks, with the observation(s) that <<4>>. You may contact your respective teacher in case of any query using the link <<5>>. Regards, Your course instructors
Sample data.txt file	i190000@nu.edu.pk, Christopher Columbus, 5, your logic is not generic, forms.google.com/oop_queries i190001@nu.edu.pk, Jonty Rhodes, 0, you run-out Inzimam in world-cups, forms.google.com/oop_queries
Output i190000@nu.edu.pk file	Dear Christopher Columbus, Find below your report of the mock final exam with the relevant observations: Q1: 5 / 10 marks, with the observation(s) that your logic is not generic. You may contact your respective teacher in case of any query using the link forms.google.com/oop_queries. Regards, Your course instructors
Output i190001@nu.edu.pk file	Dear Jonty Rhodes, Find below your report of the mock final exam with the relevant observations: Q1: 0 / 10 marks, with the observation(s) that you run-out Inzimam in world-cups. You may contact your respective teacher in case of any query using the link forms.google.com/oop_queries. Regards, Your course instructors
Explanation: The above *body.txt* and *data.txt* files should generate M=2 files (one each for Christopher Columbus and Jonty Rhodes), replacing <<2>>, <<3>>, <<4>>, and <<5>> with the corresponding name, score, comment, and link for each of the recipients.	