Bahria University, Islamabad Campus

Department of Computer Science

Final Examination

Class/Section: BSCS-2(A, B)

(Fall 2021 Semester)

Paper Type: Descriptive

Course:	Object Oriented Programming	Date: 07-02-2022
Course Code:	CSC-210	Time: 13:00
Faculty's Name:	Imran Siddiqi	Max Marks:50
Time Allowed:	2.5 Hours	Total Pages: 4

INSTRUCTIONS:

- Attempt all three questions.
- Do not write anything on the question paper except your name and enrolment number.
- Comprehension of questions is part of examination.

Student's Name: _____ Enroll No: _____
(USE CAPITAL LETTERS)

Question # 1 (10 Marks)

i. Given a class **Cube**, write the declaration/prototype of a (non-member) function to overload the "~" operator to compute the volume of the cube. (ONLY prototype is required). (2)

ii. Given "**fin**" being an object of class **ifstream** to read objects of a class **Point** from a file, write a statement that moves the file pointer associated with "**fin**" to the 5th record in the file. (2)

iii. Consider the following class **Point**.

```
class Point{
private:
      int x;
      int y;
};
```

Rewrite the class using templates so that data types of members **x** and **y** can be controlled. Both **x** and **y** can be of same or different data types.

(2)

iv. Write the output of the following statements.

```
string s1 = "abc@xyz.com";
int index = s1.find("@");
string a = s1.substr(0, index);
string b = s1.substr(index + 1);
cout << a << endl;
cout << b << endl;
```

(1)

v. State the three keywords used with exception handling in C++. (1)

vi. `cin` is an object of class _____.

Question # 2 (15 Marks)

Consider the following class **`Patient`** that stores the ID, body temperature and heart rate of each patient. Constructor and few member functions are also provided in the class.

```
class Patient{
private:
      int ID;                    //ID of the patient
      float temperature;         //Body temperature of patient
      int heartRate;             //Heart rate of patient in beats per minute
public:
      Patient(int ID = 0, float temp = 98, int heartRate = 72){
            this->ID = ID;
            this->temperature = temp;
            this->heartRate = heartRate;
      }
      void printData(){
            cout << "Patient ID:" << ID << endl;
            cout << "Temperature:" << temperature << endl;
            cout << "Heart Rate:" << heartRate << " beats per minute."<< endl;
      }
      int getID(){ return ID; }
      int getTemp(){ return temperature;}
      int getHeartRate(){ return heartRate; }
};
```

(5)

i. Using a friend function, overload the "<<" operator to display the data members of the **`Patient`** class. Given two objects **`p1`** and **`p2`** of class **`Patient`**, the operator should allow statements like **`cout<<p1<<p2;`**

(4)

ii. Using a member function, overload the "<" operator for the **Patient** class. The operator should compare the IDs of two patients.

(6)

iii. Assume objects of class **Patient** are written to a file "**Patients.dat**" in the binary mode. Complete the following global (non-member) function which accepts the total number of **Patient** objects in the file and prints the data of all patients whose body temperature is more than 99 or, whose heart rate is less than 60 beats per minute.

```
void displayPatientData(int numberOfRecords){ }
```

Question # 3 (25 Marks)

Consider the following class **LoanApplicant** that stores data of an applicant who has applied for a loan. The class includes amount of loan, start date of loan, total number of months in which the loan will be paid back and the per month salary of the applicant. (Note: Assume the class **Date** is already defined with all standard functions).

Two types of loan applications can be made, to construct a home and to purchase a car. Home loans also include information on the total value of the land on which the construction will be done while car loans include information on the model of the car to be purchased. For simplicity, it is assumed that there is no interest on any loan and applicant pays back only the amount that was borrowed.

```
class LoanApplicant{
protected:
      int loanAmount;   //Amount of loan
      Date startDate;   //Start date of the loan-Assume Date class is defined
      int months;       //Duration of loan in months
      int salary;       //Per month salary of applicant

public:


};

class HomeLoanApplicant :public LoanApplicant{
private:
      int landValue; //Value of land on which construction is to be done
public:

};

class CarLoanApplicant :public LoanApplicant{
private:
      int modelYear; //The year of the model of the car like 2020, 2021
public:


};
```

Enrollment Number: _____

Add the following functionality to the above classes.

i. A parameterized constructor in each of the three classes.

ii. A pure virtual function `print()` in the class `LoanApplicant`. The function must be overridden in each of the child classes and should print all the data related to a loan application.

iii. A pure virtual function `bool isApproved()` in the class `LoanApplicant` with functionality in the child classes. A home loan is approved if the land value is greater than the loan amount and the salary of the applicant is at least 50 times the amount of loan. Likewise, a car loan is approved if the car (model) is older than the year 2020 and the salary of the applicant is 20 times or more of the loan amount.

iv. In the main program, define an array of two pointers to the class `LoanApplicant`. Using dynamic memory allocation, create an object each of the two child classes and call the `print()` method of each object.

End of Question Paper
