|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **National University of Computer and Emerging Sciences, Lahore Campus** | | | | |
| **Course:** | **Computer Programming** | **Course Code:** | **CS-103** |
| **Program:** | **BS (Computer Science)** | **Semester:** | **Spring 2017** |
| **Duration:** | **150 Minutes** | **Total Marks:** | **50** |
| **Paper Date:** | **4 May 2017** | **Weight** | **40%** |
| **Section:** | **R** | **Page(s):** | **3** |
| **Exam:** | **Final Term** | **Reg. No** |  |

**Instruction/Notes:**

1. Understanding the question paper is also part of the exam, so do not ask any clarification.
2. Make sure to switch off your mobile phones before the Exam starts.
3. No USB’s are allowed. Please see that the area in your threshold is clean. You will be charged for any material which can be classified as ‘helping in the paper’ found near you.
4. Talking/Discussion is not allowed. It is your responsibility to protect your code and save it from being copied. If you don’t protect it all matching codes are considered copy/cheating cases.
5. You are not allowed to use internet for any purpose.
6. Submission will be on slate, file name should be your roll no. Instructor will be not responsible for any lost submission

**Question # 1 (10)**

Write template/s for this given main, do not write extra classes or functions, every extra function will deduct your marks.

|  |  |
| --- | --- |
| int main()  {              int i = 2;                  double d = 4.5;                  char c = 'e';                  Storage<int> iObj(i);                iObj.print();     //2                  Storage<double> dObj(d);                dObj.print();     //4.5                  Storage<char> cObj(c);                cObj.print();     //e                  int \*iptr = new int(2);                  double \*dptr = new double(4.5); | char \*cptr = new char('e');                  Storage<int\*> iPtrObj(iptr);                  Storage<double\*> dPtrObj(dptr);                  Storage<char\*> cPtrObj(cptr);                iPtrObj.print();     //2                dPtrObj.print();     //4.5                cPtrObj.print();     //e  } |

**Question # 2 (20)**

**Part 1 PersonData and CustomerData classes**

Design a class named PersonData with the following member variables:

**•** lastName

**•** firstName

**•** address

**•** city

**•** state

**•** zip

**•** phone

Write the appropriate getters and setters for these member variables. Next, design a class named CustomerData, which is derived from the PersonData class. The CustomerData class should have the following member variables:

• **customerNumber**

**• mailingList**

The customerNumber variable will be used to hold a unique integer for each customer. The mailingList variable should be a bool. It will be set to true if the customer wishes to be on a mailing list, or false if the customer does not wish to be on a mailing list. Write the appropriate getters and setters for these member variables.

**Part 2 PreferredCustomer Class**

A retail store has a preferred customer plan where customers may earn discounts on all their purchases. The amount of a customer’s discount is determined by the amount of the customer’s cumulative purchases in the store.

**•** When a preferred customer spends $500, he or she gets a 5% discount on all future purchases.

**•** When a preferred customer spends $1,000, he or she gets a 6% discount on all future purchases.

Design a class named PreferredCustomer, which is derived from the CustomerData class you created in part 1. The PreferredCustomer class should have the following member variables:

**• purchasesAmount (a double)**

**• discountLevel (a double)**

The purchasesAmount variable holds the total of a customer’s purchases to date. The discountLevel variable should be set to the correct discount percentage, according to the store’s preferred customer plan.

Write appropriate member functions for this class.

Make a main program to test it.

*Input Validation: Do not accept negative values for any sales figures.*

**Question # 3 (10)**

Write a recursive code that receive an integer and determine it’s an prime number or not if prime then return false else true

Bool isPrime(int num) // Do not change prototype,

(Hint: You can use one of your previous concept)