Câu 1

**Edit: Design and code a class name DessertIterm that holds information of a DessertIterm**

**Design and code a deringving class name Candy from DessertIterm that holds information about Candy**

|  |
| --- |
| **DessertIterm** |
| *String Name;* |
| *DessertIterm( String Xname)* |

|  |
| --- |
| **Candy** |
| *Price doublle;*  *Weight double ;* |
| ***Candy****( String Xname, double X;price, double Xweight)*  **double getCost()** |

**Constructers to set value for instance variables.**

*Add needed operations to the class so that the main function can be run and complete the below method which is declared in Candy class, the function will be used in second test case*

In Candy class: -double **getCost**()-use to determine its cost, cost=price\*weight, the cost shoude be arounded to the closest double value with tie rounding up if Candy name start with letter “C” or”c”, ortherwisre the cost shoude be a rounded to the closest double value with ties rounding down

**Test Case**

*Enter dessert name: Whilte Chocolate Fudge*

*Enter candy price: 1.04*

*Enter candy weight: 2*

*Enter TC: 2*

*OUTPUT:*

*2.0*

*Enter dessert name: Chocolate Fudge*

*Enter candy price: 1.04*

*Enter candy weight: 2*

*Enter TC: 2*

*OUTPUT:*

*3.0*

*…………………………………………………*

*Enter dessert name: Chocolate Fudge*

*Enter candy price: 1.04*

*Enter candy weight: 2*

*Enter TC: 1*

*OUTPUT:*

*Chocolate Fudge*

*Chocolate Fudge 2.0 1.04*

Câu 2.

Edit:

|  |
| --- |
| Iinvoice (interface)-do not edit |
| + String f1(ArrayList<invoice a, int st>;  ++ int f2(ArrayList<invoice a); |

|  |
| --- |
| Invoice |
| - String Name  - int Total amount; |
| -Invoice(String Name, int Totalamount) |

|  |
| --- |
| **MyInvoice** |
| + String f1(ArrayList<invoice a, int st>  **+** **int f2(ArrayList<invoice at>**    f2(ArrayList<invoice a) |

**+**

**Constructers to set value for instance variables.**

String f1(ArrayList<invoice a, int st>-if st=1, firstly, sort the list “a” of invoices by customer name ( *ascending*), finally, return customer name of second invoice in the list “a”. The must ignores the case during comparison.

if st=2, firstly, sort the list of invoices by total amount (*ascending*) **,** finally, return customer name of second invoice in the list “a”.

int f2(ArrayList<invoice a> : set x the first highest amount of invoice and y is the second amount of invoice;remove all invoice wich have the amount of invoice equals to amount of x and y in the list “a”, finally caculate and return the sum of amount of all ather invoice in the list a;

*Enter number of Invoice: 4*

*name to customer: Noi*

*Total amount: 96*

*name to customer: Ha*

*Total amount: 77*

*name to customer: Mua*

*Total amount: 96*

*name to customer: nay*

*Total amount: 100*

*Enter test function (1-f1;2-f2): 1*

*OUTPUT:*

*ST=1*

*Mua*

*Enter number of Invoice: 4*

*name to customer: Ha*

*Total amount: 96*

*name to customer: Noi*

*Total amount: 77*

*name to customer: Mua*

*Total amount: 96*

*name to customer: nay*

*Total amount: 100*

*Enter test function (1-f1;2-f2): 1*

*OUTPUT:*

*ST=2*

*Ha*

*Enter number of Invoice: 4*

*Issue to customer: Obama*

*Total amount: 96*

*Issue to customer: Putin*

*Total amount: 77*

*Issue to customer: Xi Jinping*

*Total amount: 96*

*Issue to customer: Trump*

*Total amount: 100*

*Enter test function (1-f1;2-f2): 2*

*OUTPUT:*

*77*

Câu 3. Design and code a class named MyCard that holds information of a payment card, including:

|  |
| --- |
| MyCard |
| *String cardNumber;*  *String type;;* |
| MyCard*( String cardNumber, String type)* |

* Constructor to set values for instance variable. Make sure that a valid payment card number is a string which constains **letters** only, reset payment card number to a string of “0000” (4(four) digits of 0) if payment card number is not valid.
* Add need operations to the class and complete below method which is declared in MyCard class, the function will be used in the second tesst case.

+ String getCardCode()- assuming that length of payment card is greater than 4 this function return value of card code as the rule:

* Card code= last 4 didits of payment card number if type of card is “credit”.
* Atherwise card code= firts 4 didits of payment card number;

**TEST CASE**

Enter card type: credit

Enter card number: hanoimuanayvangnhungconmua

Enter TC: 2

OUTPUT:

nmua

Enter card type: credit

Enter card number: hanoimuanayvangnhungconmua

Enter TC: 1

OUTPUT:

credit hanoimuanayvangnhungconmua

Enter card type: debit

Enter card number: hanoimuanayvangnhungconmua

Enter TC: 2

OUTPUT:

Hano

Enter card type: debit

Enter card number: hanoi7muanayvangnhungconmua

Enter TC: 1

OUTPUT:

debit 0000