Homework Assignment 5

Due Date: 18:30, May 12, 2016

1 Problem 1

For this homework assignment, you write a program about the checkout bill of Dessert Shop which sells candy by the gram, cookies by the dozen, ice cream with or without topping. Your program will be used for the checkout system. To do this, you will implement an inheritance hierarchy of classes derived from a `DessertItem` abstract base class. The `Candy`, `Cookie`, and `IceCream` classes will be derived from the `DessertItem` class. The `Topping` class will be derived from the `IceCream` class. You will also write a `Checkout` class which maintains a list (`list`) of `DessertItem` pointers.

p1.hpp

```
#include <iostream>
#include <string>
/*add the header if you need*/
using namespace std;

//Abstract base class for Dessert Item hierarchy
class DessertItem
{
private:
//Name of the DessertItem object
    string name;

public:
    DessertItem(){}
    DessertItem(string name):name(name){}
//Empty virtual destructor for DessertItem class
    virtual ~DessertItem(){}
//returns Name of DessertItem
    string getName(){ return name;}
    virtual string getDetails() = 0;
    virtual double getCost() = 0;
};

class IceCream : public DessertItem
{

    /* Write about IceCream Constructor
       IceCream(string name, double cost):DessertItem(name),cost(cost)
    */

    /* Write about IceCream other member functions*/
private:
    double cost;
};

class Topping : public IceCream
{
```

```
    /* Write about Topping Constructor
       Topping(string iceCreamName, double iceCreamCost,
        string toppingName, double toppingCost)
    */

    /* Write about Topping other member functions*/
private:
    string toppingName;
    double toppingCost;
};

class Cookie : public DessertItem
{
    /* Write about Cookie Constructor
       Cookie(string name, int number, double pricePerDozen)
    */

    /* Write about Cookie other member functions*/

private:
//Number of dozens of Cookie
    int number;
    double pricePerDozen;
};

class Candy : public DessertItem
{
public:
    /* Write here about Candy Constructor
       Candy(string name, double weight, double pricePerGram)
    */
    /* Write about Candy other member functions*/
private:
//Weight of Candy
    double weight;
    double pricePerGram;
};



class Checkout {

    friend ostream &operator<<(std::ostream &, Checkout &);


    /* Write about Checkout member functions
    1. "enterItem" function to add the element into the list
    2. "removeItem" function to remove the elemtent from the list
    3. calculate the total cost and tax in the list
    4. "numberOfItems" for number of Item in the list
    5. "clear" clear all Items from list
    */
private:
    list<DessertItem*> itemList;
};
```

```
ostream &operator<<(ostream &output, Checkout &checkout){

/*Overloaded operator that output a receipt for the current list of items*/

}
```

main.cpp

```
#include<iostream>
#include"p1.hpp"
using namespace std;

int main()
{
        Checkout checkout;
        Candy candy("Peanut Butter Fudge",1020,0.26);
        checkout.enterItem(&candy);
        IceCream icecream("Vanilla Ice Cream",35);
        checkout.enterItem(&icecream);
        Topping topping("Choc.ChipIce Cream",45,"Crisps",15);
        checkout.enterItem(&topping);
        Cookie cookie("Oatmeal Raisin Cookies",4,99);
        checkout.enterItem(&cookie);
        cout<<checkout<<endl;
        checkout.clear();
        return 0;
}
```

1.1 Sample Output

```
Welcome to OOP's shop
------------------------------

Number of items: 4

Peanut Butter Fudge                      265
(1020 gram(s) * 0.26/gram)
Vanilla Ice Cream                         35
Crisps Sundae with Choc.ChipIce Cream     60
Oatmeal Raisin Cookies                   396
(4 dozen(s) * 99/dozen)

------------------------------
Cost                                     756
Tax                                       38

Total cost                               794
```

2 How to submit the assignment?

1. Name the source code of each problem as following:
   - Prblem1: p1.hpp
2. Do not rename the files or put them into any directory. Upload them directly to the **e-Campus (E3)** system. You will get no credit if you don't follow the rule. Note that the penalty for late homework is **15% per day**, and late homework will not be accepted after 3 days past the due date. In addition, homework assignments must be individual work. If I detect what I consider to be intentional plagiarism in any assignment, the assignment will receive **zero credit.**