```
/*
A common place to buy candy is from a machine. The machine sells candies,
chips,
gum, and cookies. You have been asked to write a program for this candy
machine.
The program should do the following:
1. Show the customer the different products sold by the candy machine.
2. Let the customer make the selection.
3. Show the customer the cost of the item selected.
4. Accept money from the customer.
5. Release the item.
The machine has two main components: a built-in cash register and several
dispensers
to hold and release the products.
Define class cashRegister in C++ with the following descriptions :
Private Members:
cashOnHand of type integer
Public Members:
A default constructor cashRegister() sets the cash in the register to 500.
A constructor cashRegister(int) sets the cash in the register to a specific
amount.
A function getCurrentBalance() which returns value of cashOnHand
A function acceptAmount(int) to receive the amount deposited by the customer
and
update the amount in the register
Define class dispenserType in C++ with the following descriptions :
Private Members:
numberOfItems of type integer
cost of type integer
Public Members:
A default constructor dispenserType () sets the cost and number of items in
the
dispenser to 50 each.
A constructor dispenserType (int,int) sets the cost and number of items in the
dispenser to the values specified by the user.
A function getNoOfItems() to return the value of numberOfItems.
A function getCost() to return the value of cost.
A function makeSale() to reduce the number of items by 1.
When the program executes, it must do the following:
1. Show the different products sold by the candy machine.
2. Show how to select a particular product.
Once the user has made the appropriate selection, the candy machine must act
accordingly. If the user has opted to buy a product and that product is
available, the
candy machine should show the cost of the product and ask the user to deposit
the
money. If the amount deposited is at least the cost of the item, the candy
machine
```

```
should sell the item and display an appropriate message.
Divide this program into three functions: showSelection, sellProduct, and
main.
The function sellProduct must have access to the dispenser holding the product
(to
decrement the number of items in the dispenser by 1 and to show the cost of
the item)
as well as the cash register (to update the cash). Therefore, this function
has two
parameters: one corresponding to the dispenser and the other corresponding to
the
cash register.
*/

#include<iostream>
#include<stdlib.h>
using namespace std;

class cashRegister
{
    private:
        int cashOnHand;

    public:
        cashRegister();
        cashRegister(int cash_in);
        int getCurrentBalance();
        void acceptAmount(int cash_dep);
};
class dispenserType
{
    private:
        int numberOfItems,cost;

    public:
        dispenserType ();
        dispenserType (int cost1,int no_items);
        int getNoOfItems();
        int getCost();
        void  makeSale();
};
cashRegister::cashRegister()
{
    cashOnHand = 500;
}
cashRegister::cashRegister(int cash_in)
{
    cashOnHand = cash_in;
```

```
}
int cashRegister::getCurrentBalance()
{
    return cashOnHand;
}
void cashRegister::acceptAmount(int cash_dep)
{
    cashOnHand = cashOnHand + cash_dep;
}

dispenserType::dispenserType()
{
    numberOfItems = 50;
    cost = 50;
}
dispenserType::dispenserType (int cost1,int no_items)
{
    numberOfItems = no_items;
    cost = cost1;
}
int dispenserType::getNoOfItems()
{
    return numberOfItems;
}
int dispenserType::getCost()
{
    return cost;
}
void dispenserType::makeSale()
{
    numberOfItems--;
}
void showSelection();
void sellProduct(dispenserType &product, cashRegister &ucash);
int main()
{
    cashRegister counter;
    dispenserType candy(100, 3);
    dispenserType gum(100, 2);
    dispenserType lolipop(75, 5);
    dispenserType chocolate(50, 2);
    int choice;
    showSelection();
    cin >> choice;
    while (choice != 5)
    {
        switch (choice)
        {
```

```
            case 1:
                system("cls");
                sellProduct(candy, counter);
                break;
            case 2:
                system("cls");
                sellProduct(gum, counter);
                break;
            case 3:
                system("cls");
                sellProduct(lolipop, counter);
                break;
            case 4:
                system("cls");
                sellProduct(chocolate, counter);
                break;
            default:
                cout << "Invalid selection." << endl;
        }
        showSelection();
        cin >> choice;
    }
    return 0;
}


void showSelection()
{
    cout<<"******MENU******";
    cout<<"\n1.CANDY\n";
    cout<<"2.GUM\n";
    cout<<"3.LOLIPOP\n";
    cout<<"4.CHOCOLATE\n";
    cout<<"5.Exit\n";
}

void sellProduct(dispenserType &product, cashRegister &ucash)
{
    int amount;
    int amount2;
    if (product.getNoOfItems() > 0)
    {
        cout << "Please deposit " << product.getCost() << " Rs" << endl;
        cin >> amount;
        if (amount < product.getCost())
        {
            cout << "Please deposit another " << product.getCost()- amount <<
" Rs" << endl;
            cin >> amount2;
```

```
            amount = amount + amount2;
        }

        if (amount >= product.getCost())
        {
            ucash.acceptAmount(amount);
            product.makeSale();
            cout << "Collect your item at the bottom and " << "enjoy." <<
endl;
        }

        else
            cout << "The amount is not enough. " << "Collect what you
deposited." << endl;
            cout << "*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*" << endl << endl;
    }
    else
        cout << "Sorry, this item is sold out." << endl;
}
```