Homework #4
Functions & Vending Machines

Assigned: September 20, 2017
Due: Sep. 27 by 11:59:59 PM

Write a C++ program which will model accepting payment and giving change for a snack vending machine. The machine will contain the following items, shown with the label used to select the item, and its price:

Label	Item	Price
P	Potato Chips	$ 1.25
S	Snickers Bar	$ 1.35
T	Pop Tart	$ 0.95
C	Cookies	$ 1.50
B	Brownies	$ 1.75
N	Nuts	$ 1.40

This assignment does have an extra credit option. Please see the **Extra Credit** section below.

Requirements:

- Name the source file for your program `program4.cpp`

- Throughout the program, all amounts of money must be stored internally as *cents* (in variables of type `int`).

- There must be three functions with the following prototypes (Choose a naming scheme that suits you, but use these names):

```
int menu(void);
int acceptMoney(int price);
int computeChange(int totalPaid, int totalPrice);
```

These three functions must behave as follows:

- The function `menu` must display the list of snack items and prompt the user for their choice. If the user makes a valid selection, the function must return the price of that selection, otherwise the user must be re-prompted for a choice until a valid choice from the menu is obtained.

- The function `acceptMoney` must give the user a choice of coins to insert, and add up the total amount inserted by the user, in a loop until this total amount is equal to or greater than the function's parameter `price` (which is the cost of the user's choice of snack). Once this loop is complete, the function must return the total amount of money inserted by the user.

– The function `computeChange` must return the amount of change owed to the user, found by subtracting `totalPrice` from `totalPaid`.

- For any menu choice where a letter is expected, if a certain letter is a valid menu choice, then accept that letter in both upper- and lower-case.

- For the menus for selecting a snack and selecting an amount of money to insert, if an invalid selection is made, then re-prompt the user for a selection until a valid selection is made. For the choice of whether to purchace another snack, 'y' and 'Y' mean yes, and anything else means no.

- The main program must contain a loop that calls the functions `menu`, `acceptMoney` and `computeChange` using suitable variables to store the amounts of money involved, then displays the total amount inserted and the change returned. After completing a transaction, it must ask the user whether they would like to make another purchase, and continue as long as the user chooses to.

- Use comments to show the main steps in the program, and to document variable declarations.

- A sample run of your program should look like:

```
Welcome to the snack vending machine

Available snacks to select from:
     P - Potato Chips    $1.25
     S - Snickers Bar    $1.35
     T - Pop Tart        $0.95
     C - Cookies         $1.50
     B - Brownie         $1.75
     N - Nuts            $1.40
Please enter the letter labeling your snack selection: P

Money accepted by the machine:
     N - Nickel
     Q - Quarter
     D - Dollar

Your selected snack item cost: 125 CENTS
Your total inserted:             0 CENTS
Insert amount (enter letter of choice): d

Your selected snack item cost: 125 CENTS
Your total inserted:           100 CENTS
Insert amount (enter letter of choice): D

Your total inserted: 200 CENTS
```

```
Dispensing change:     75 CENTS

Would you care to make another purchase (Y/N): y

Available snacks to select from:
     P - Potato Chips    $1.25
     S - Snickers Bar    $1.35
     T - Pop Tart        $0.95
     C - Cookies         $1.50
     B - Brownie         $1.75
     N - Nuts            $1.40
Please enter the letter labeling your snack selection: k
Invalid selection!

Please enter the letter labeling your snack selection: n

Money accepted by the machine:
     N - Nickel
     Q - Quarter
     D - Dollar

Your selected snack item cost: 140 CENTS
Your total inserted:             0 CENTS
Insert amount (enter letter of choice): d

Your selected snack item cost: 140 CENTS
Your total inserted:           100 CENTS
Insert amount (enter letter of choice): Q

Your selected snack item cost: 140 CENTS
Your total inserted:           125 CENTS
Insert amount (enter letter of choice): w

w is not recognized as a coin.

Your selected snack item cost: 140 CENTS
Your total inserted:           125 CENTS
Insert amount (enter letter of choice): Q

Your total inserted: 150 CENTS
Dispensing change:    10 CENTS

Would you care to make another purchase (Y/N): n

Thank you and enjoy your purchase!
```

Hints:

- The function `setw` from the library `<iomanip>` may be useful.

- The function `toupper` from the library `<cctype>` may be useful.

- While three functions are required, you are not restricted to three functions.

Extra Credit:

- For 2 points (5%) extra credit on this assignment, instead of showing only the value of the change owed to the user, dispense the least amount of coins necessary to make the change.

- It should look something like this:

```
Your selected snack item cost: 135 CENTS
Your total inserted:             0 CENTS
.
// THIS IS WHERE THE USER INSERTS THEIR COINS
.
Your total inserted: 200 CENTS
Dispensing change:      2 QUARTERS
                        1 DIME
                        1 NICKEL
Total change:          65 CENTS
```

Reminders:

- Be sure that your program includes your name, ID, description, etc. as shown in the General Homework Requirements Handout

- Use good style including indentation, comments, etc. Part of the grade will be for style and quality.

- Carefully test your program.

- You are welcome to write your program at home. If you do, be sure to compile and test it in the lab before submitting it.

How to submit your program:

- Submit the file `program4.cpp` electronically using `~cs211a/bin/handin 4 program4.cpp`