CSCI 272 Spring 2019 **Programming Project 2**

Project Description: Complete the definitions of the functions for the ConcessionStand class in the ConcessionStand.cpp file. The class definition and function prototypes are in the provided ConcessionStand.h header file. A testing program is in the provided main.cpp file. You don't need to change anything in ConcessionStand.h or main.cpp, unless you want to play with different options in the main.cpp program.

Student Collaboration: You're welcome to discuss labs and projects and share ideas about how functions might be implemented. However, do not copy code from others or give your code to others. Project submissions that are virtually identical, including insignificant changes like different variable names, may receive a 0.

Submission: Submit the completed **ConcessionStand.cpp** file and a screenshot of your output on Blackboard.

Due: Friday, March 8, 2019, end of day.

The `ConcessionStand` class has the following:

private data members that are only accessible by the class member functions:

`standName`—the name of the concession stand (e.g. Pizzeria, Candy etc.)

Three parallel vectors to hold product name, price and number available data:

`products`—a vector to hold product names
`prices`—a vector to hold product prices
`quantity`—a vector to hold the number of each product available for sale

In parallel vectors, [0] in each vector refers to the same product, [1] in each vector refers to the same product, and so on. For example, the price for products[0] is in prices[0].

publicly available functions:

`ConcessionStand` default constructor—This is defined in ConcessionStand.h, you don't need to define in ConcessionStand.cpp

`ConcessionStand` custom constructor—Takes in a string value to initialize `standName`

`displayMenu`—Displays each product name in the products vector along with its associated price. Display dollar amounts with 2 decimal places and trailing zeros. If the quantity for an item is 0, display "SOLDOUT" instead of the price.

```
MENU
----
Pizzeria
1. Cheese     $ 3.50
2. Veggie     $ 4.50
3. Pepperoni  $ 4.50
4. Mushroom   $ 4.50
```

```
MENU
----
Pizzeria
1. Cheese     SOLDOUT
2. Veggie     $ 4.50
3. Pepperoni  $ 4.50
4. Mushroom   SOLDOUT
```

`printInventoryToFile`—Print to a file (e.g. “inventory.txt”) each product name and associated quantity. This function will return a value of false if the output file could not be opened, otherwise it will return a value of true. For example,

inventory.txt saved

```
1  INVENTORY
2  =========
3  Cheese     0
4  Veggie     2
5  Pepperoni  3
6  Mushroom   0
7
```

`addItemToMenu`—This function takes in a string for product name, double for product price, and unsigned int for the product quantity. It will add the product name, price and quantity to the appropriate parallel vector.

`sellMenuItem`—This function takes in the number from the menu that the customer selected. For example, if the customer chose 1, that would mean a sale of Cheese pizza in the example above. *Keep in mind though, that while the menu is numbered 1 – 4, the index values in the vector will be 0 – 3.* Before selling the item, make sure it is not sold out. If it is sold out, display a message. Otherwise, decrease the quantity for the sold item.