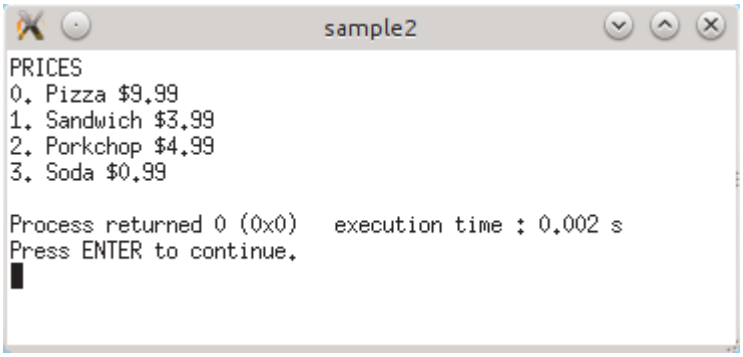
Class 3 Homework

If Statements, Classes, and Functions

1. Expand the "Prices" program

In the handout, we work on the prices program.



In the program from the handout, we have a ShopItem class and we store a price and name for our items.

Then we store an array of ShopItems, and iterate through them with a for loop to show the user a list of items and prices.

Here are some ways you can expand this program:

Selecting Items
Let the user select items (by entering a menu number), and as they select items, a variable storing a "total price" is added to.
You will also need a loop in main to make sure the game doesn't quit until the user selects a "quit" option from the menu.

Select Multiple Items In One Go
In addition to the feature above, after the user chooses an item, ask them how many of that item they would like. Then, add the price multiplied by the quantity to get the amount to add to the "total price" variable.

Refactoring!
Main shouldn't be the place in your program to store the program logic.
Create a "Store" class (in its own .h and .cpp files) to encapsulate all of our ShopItems and buying abilities.
Have your resulting program look something like this:

main.cpp

```
1  #include <iostream>
2  #include "Store.h"
3  using namespace std;
4
5  int main()
6  {
7      Store cppStore;
8
9      while ( cppStore.DoneWithStore() == false )
10     {
11         cppStore.ListOptions();
12         cppStore.HandleUserChoice();
13     }
14
15     cout << endl;
16     cout << "CHECKING OUT YOUR GROCERIES..." << endl;
17
18     cout << "Total price:\t\t $" << cppStore.TotalPurchasePrice() << endl;
19     cout << "\n\nThank you for shopping at CPP SHOP" << endl;
20
21     return 0;
22 }
```

This version now has two classes – Store and ShopItem. The Store has an array of ShopItems.

2. File Output

Write a program that asks the user for pizza toppings. Each toping costs extra.
Every time the user selects another pizza topping, output to the text file the topping name and the price of it.
At the end of the program, output the total cost to the file and close the file.