CSci 1113: Introduction to C/C++
Programming for Scientists and Engineers
Homework 9
Fall 2018

Due Date: Wednesday, December 5, 2018 before 11:00pm.

Instructions: This is an individual homework assignment. There are two problems worth 20 points each. Solve the problem below by yourself (unlike the labs, where you work collaboratively), and submit the solution as a C++ source code file. Here are a few more important details:

1. Unlike the computer lab exercises, this is not a collaborative assignment.

2. Because all homework assignments are submitted and tested electronically, the following are important:
   - You follow any naming conventions mentioned in the homework instructions.
   - You submit the correct file(s) through Moodle by the due deadline.
   - You follow the example input and output formats exactly given in each problem description.
   - **Regardless of how or where you develop your solutions, your programs compile and execute on cselabs computers running the Linux operating system.**

3. You should test your program on other test cases (that you make up) as well. Making up good test cases is a valuable programming skill, and is part of ensuring your code solution is correct.

Problem A: Crafting products (20 points)
Your program first ask which file to read. This will contain three commands:

- Item: -- This command is used to add a new item to a list. It will provide both the item name and the price of this item.
- Recipe: -- This command defines and item that is built from other basic items. The price of this item will be the sum of the individual components to make the item. It will be a list of names of other items (already defined) that it is made from.
- Change: -- This will change price of an already defined item using the “Item:” command. It will not be used to redefine “recipes”. This will provide both the name of the item being redefined and the new price.

After all the items have been entered. For each “Item” and “Recipe” show the price to get this item. For part A, you can assume recipes will only be used from simple “Items” instead of more complex “Recipes”. The price changes from the “Change” command should be reflected on both the simple “Item” and in the “Recipes” that use the item. In other words, the change in the price should effect all products.

You may assume:

- Item names will be single words.
- One command per line.
- Commands will be formatted as in the file (space after the command and no trailing space).
- “Recipes” will only use “Items” that have already been defined.
- “Change” will only modify “Items” that have already been defined.
- Recipes will not be more than 50 items.
- There will not be more than 500 commands in the file.

- Products (both “Items” and “Recipes”) are unique. In other words, there will never be two products with the same name.

Hint: It might be useful to think about how you want to store data and handle changes efficiently before starting this part. In the next part you will have to handle greater changes, which can be done easily if you use the correct approach.

Example 1 (user input underlined):
```
What file to open?
list.txt
wood : 2.5
nail : 1.5
tape : 2
nailbat : 7
shuriken : 8
walkingstick : 7
spear : 7.5
```

Example 2 (user input underlined):
```
What file to open?
listA.txt
wood : 2.5
nail : 1.5
tape : 4
nailbat : 7
shuriken : 10
walkingstick : 9
spear : 9.5
```

When you are done, name the source code file <username>_9A.cpp. Here you replace <username> with your U of M email address; for example, if your email address is smithx1234@umn.edu, your file should be named smithx1234_9A.cpp. Then submit your program using the HW 9 Problem A submission link in Moodle.

Problem B: Crafting MOAR products (20 points)
This part extends your answer to part A. In addition to the three commands outlined above, you also need to incorporate one new command and modify one:

- Recipe: -- This command is still used to specify a list of items from which you can build it, except it can now over other recipes as part of them.
- Modify: -- This command changes the recipe list for an already defined item. This is very similar to the “Change” command, except for recipes.

Again, the output should simply be the list of items and their prices.

Example 1 (user input underlined):
```
What file to open?
listB.txt
wood : 1
nail : 1.5
tape : 4
nailbat : 4
shuriken : 10
```

```
walkingstick : 3.5
spear : 8
stilts : 15
elevatedhouse : 67
```

When you are done, name the source code file <username>_9B.cpp. Here you replace <username> with your U of M email address; for example, if your email address is smithx1234@umn.edu, your file should be named smithx1234_9B.cpp. Then submit your program using the HW 9 Problem B submission link in Moodle.