CS 360 100P
Making Change

Revision Date: January 10, 2013

Topics to learn

Please read up on the following subjects:

- dynamic programming
- recursion with memoization

Tasks to perform

Your task is to determine the number of unique ways to make up a total amount in change using a specified set of coin denominations. For example, suppose you need to return six cents in change, using U.S. coinage. There are two ways to do this: one nickel and one penny or six pennies. Realize, that when making change, you can either use a coin or not. So just count the ways to make the remaining amount of change if you use a coin and add that to the number of ways to make change if you don't use the coin.

You are to implement three versions of a change-making algoirthm, one which uses dynamic programming, one which uses recursion plus memoization, and one which just uses recursion. You should first write the purely recursive version. Then you should add memoization. The memoized version will give you a a hint on how to do the dynamic programming version.

You must implement this program using the Java programming language. You must provide a makefile.

Testing

You should name your program *make-change*. Your program should take command-line arguments so that the type of computation, the total, and the coin set can be entered easily. For example, the command:

```
java make-change -d 68 50 25 10 5 1
```

specifies the example used above and should print 6, using dynamic programming. A `-m` option means to use recursion plus memoization. A `-r` option means recursion without memoization.

Other concepts

Be able to describe to your mentor the complexity of your three versions.