

CS 215 - Fundamentals of Programming II
Spring 2017 - Homework 2
10 points

Out: January 18, 2017
Due: January 23, 2017 (Monday)

Write a program in a file named **change.cpp** that will read a series of positive integers each representing a number of cents from a file. For each integer, the program will compute the minimum number dollars, quarters, dimes, and pennies that would be needed to give out the amount in change. (Do not use nickels.) These results are to be output to a file in tabular format as shown below with each field having a width of 10.

The program **must be implemented** using a function **compute_change** that *receives* a positive integer representing a number of cents and *passes back* the number of dollars, quarters, dimes, and pennies. A function prototype must be written above the main program and the function definition must be written below the main program. Here are some notes that may be helpful.

- The change should be computed from dollars to pennies, i.e. largest to smallest.
- The change for each denomination can be computed using integer divide and remainder (%).
- Value parameters are copies of the actual argument, so they can be used like local variables.

The program must accept the input and output file names as **command-line arguments**. Proper error checking of the number of command-line arguments and successful file opens must be made.

If the input file is named **input.dat** and contains the following data:

186
247
63

Then the program would be compiled and run using:

```
$ clang++ change.cpp -Wall -o change
$ ./change input.dat output.dat
```

And the output file named **output.dat** should contain the following results:

Amount	Dollars	Quarters	Dimes	Pennies
186	1	3	1	1
247	2	1	2	2
63	0	2	1	3

What to submit

The program file must have a comment at the top of the file with your name and assignment number.

Electronically submit a **tarfile** containing **change.cpp** as explained in the handout [Submission Instructions for CS 215](#) by 11:59pm on Monday, January 23, for full credit.