

# CS 100 Project One – Spring 2019

**Project Overview:** A business owner is unhappy with the services provided by the government and has chosen to pay the tax using a combination of coins including quarters, dimes, nickels, and pennies. In this project, you are asked to write a program to analyze the tax payment in coins. The program first asks for the number of quarters, dimes, nickels and pennies the owner has paid. It then computes the total amount of tax paid in dollars and cents, and computes the weight of all the coins in pounds. Next assume the same amount of tax will be paid using paper bills in the following denominations: \$20 bills, \$10 bills, \$5 bills, and \$1 bills. (If the tax amount contains any cents, round it to the nearest dollar.) The program will figure out how to pay the tax with the minimal number of bills (therefore the minimal weight), and compute the weight of these bills in both grams and pounds.

When implementing your program, please use the following facts.

- The weight of coins: quarter (5.67 grams), dime (2.268 grams), nickel (5.0 grams) and penny (2.5 grams) based on <https://www.usmint.gov/learn/coin-and-medal-programs/coin-specifications>.
- According to the Bureau of Engraving and Printing, all U.S. bills weigh one gram.
- When converting weight from grams to pounds, use 1 gram = 0.00220462 pounds.
- Whenever trying to convert a real number into an integer, use the **round** function in **math.h** to convert. For example, to convert 2.51, use **(int)round(2.51)** that gives 3.

The following is an example execution of the program.

```
Enter the number of quarters: 44444
Enter the number of dimes: 33333
Enter the number of nickels: 22222
Enter the number of pennies: 11111

Total payment is 15666 dollars and 51 cents.
Total weight of all the coins is 1028.420450 pounds.
It is recommended to use the following payment form:
    783 twenty-dollar bills
    0 ten-dollar bills
    1 five-dollar bills
    2 one-dollar bills
It achieves the minimal weight of 786 grams or 1.732831 pounds.
```

## What You Need To Do

- Create a directory **project1** on your machine. In that directory, create a file named **payment.c**
- In **payment.c**, write the code needed to solve the problem stated above. Make sure that your program:
  - Has a header block of comments that includes your name and a brief overview of the program.
  - Reads four integers. You can assume that all input for this program will be legal (non-negative integers). Do not modify the expected input format or order.
  - Computes and prints the expected results.
- Please compile the program using **gcc -Wall payment.c -lm**
- Once you have a working program, enter some sample data to test it. You are allowed to post the results on Piazza to see whether others agree with your results. However, posting any part of C code from the project on Piazza is prohibited.
- When you are ready to submit your project, compress your **project1** directory into a single (compressed) zip file, **project1.zip**. See the **Basics** document on Blackboard if you don't remember how to do it.
- Once you have a compressed zip file named **project1.zip**, submit that file to Blackboard.

**Project 1 is due at 5:00pm on Friday, January 25. Late projects are not accepted.**

**This document including its associated files is for your own personal use only.**

**You may not post this document or a portion of this document to a site  
such as chegg.com without prior written authorization.**

**A project shall be completed individually, with no sharing of code or solutions.**

**All submissions will go through MOSS (Measure Of Software Similarity)  
for similarity check.**

**The University of Alabama's Code of Academic Conduct will be rigorously enforced.**