

CS& 131

Project 1: "Change Counter"

Name:

Since programmers are supposed to be able to make big bucks your first task is to create a program to deal with money.

Design, write and test a fully documented, user friendly ANSI C program which takes in (a) a dollar amount for a purchase and (b) the amount tendered for the purchase. The program should then give a listing of the change, specifically in the denominations listed below.



To do this, calculate the total change as an *integer* number of cents. It should be accurate for a positive, zero or negative amount of change.

Enter in the cost and tendered amounts as *fixed floating point* numbers, rounding off to the nearest cent--**be careful not to lose pennies due to rounding errors**. Print these values to two places. Calculate the total change as a *floating point* number, round off to the nearest cent, and display it to two decimal places. Now use an integer variable to store this number (multiplied by 100) as the number of cents. Divide to find the number of twenties, tens, etc., exactly as in the example below--no other denominations. You **MUST** use the % (or %=) operator to find the remainder after each denomination is removed from the total. Do not use arrays or loops.

A run of the program might look like this (**Bold Green Numbers**) are those entered by the user):

Welcome to Change Counter by <your name>!

```
Please enter the total amount of purchase: $52.173
$52.17
Please enter amount of money tendered: $60
$60.00
Your change is: $7.83
-----
Twenties : 0
Tens      : 0
Fives     : 1
Ones      : 2
Quarters  : 3
Dimes     : 0
Nickels   : 1
Pennies   : 3
-----
Thank you for using Change Counter!
```

The program should be fully planned *in advance*. It should be well documented, and work correctly.

Make sure to:

- Use *constants* and *variables* to represent **ALL** the relevant values.
- Select reasonable identifier names such as *numberOfDimes*, *totalChangeInDollars*, *changeInCents*, etc.
- If the amount tendered is less than the cost, the change should represent the amount still owing.
- Check your answers to see if they are reasonable. Watch for and test for rounding errors.
- Check to see that your heading block is completely filled in.
- Make sure that your code matches the department style guidelines.

Testing Information:

This program should work for any numbers entered, not just those shown in the example.

To test this program, try various inputs, making sure that all denominations are tested. Make sure to use inputs that might cause problems with the program (boundary tests), testing positive, zero and negative. **Make sure to check the program's answers against the *right* answers -- you may use the sample program to see what these are!**

When you are sure that the program is working correctly, solve the following problem:

- Calculate how much money is spent on gasoline each year in the U.S.
- **Use the program to:**
 - buy that much gas
 - use the U.S.'s **G.D.P.** (*Gross Domestic Product*) to pay for it (to test total accuracy, if the G.D.P. is \$13 trillion, enter 13000000000000.03)
 - **see how much we have left over -- compare the numbers from the program to hand calculations for the same numbers.**
Make sure to show expected and actual values for all calculated numbers (down to the penny), that is, the total change and **all denominations.**
Include this test in your test plan.
- What percentage of our **G.D.P.** do we spend on gasoline in a year currently?

To help you do this, fill in the chart below:

For each of the following, record in the space below a reasonable estimate of the stated factor, and describe the source of each estimate:

Estimates for the year _____ (*specify*)

Factor	Estimate	Source of Your Estimate
Gross Domestic Product of the U.S.		
Number of vehicles on the road in the U.S.		
Average Miles driven per vehicle per year		
Average MPG for all vehicles on the road		
Average price per gallon of vehicle fuel		
Calculations:	Answer	Formula used
Money spent on motor vehicle fuel that year in the U.S.		
Percentage of GDP spent on motor vehicle fuel in the U.S.		

Deliverables:

Physical:

The Project should be turned in inside a clear plastic file folder. This folder should have a simple flap to hold paper in place--NO buttons, strings, Velcro, etc. Pages should be in order, **not** stapled.

- ☐ Assignment Sheet (*printed pdf from the web*), with your **name** on it, as a cover sheet.
- ☐ Printed Source Code with Comments (*including heading blocks. Describe parameters, no line wrapping*)

Electronic:

- ☐ All source (.c) and .exe (Release Version) files
- ☐ Part 1B-J (*the printer-friendlier form*), completely filled out. Submit the .pdf file.
- ☐ A file (.rtf) containing of a test run of your program's output -- run the program, copy the window using **<ALT|PRTSCN>**, paste into Paint, invert colors (**<Ctrl|Shift|I>**), copy, open Wordpad, save.
- ☐ **It is very likely** that you *sometimes* get a "*wrong*" answer for your program output.. If your program output does not always agree with your check, **write a paragraph** containing a possible explanation for the discrepancy.
- ☐ A completed test plan, including the gasoline/G.D.P test described above. Make sure the tests include *both* expected and actual values for **all** calculated items (*i.e., twenties, tens, fives, etc.*)
- Zip all of these files together. Do not use rar or any archive format other than zip. Rename the file: "<YourName>_p1.zip".
- Submit this single zip file by going to **Canvas**, select this class, select the **Assignment** tab on the left, select **Assignment 1**, select the **submission** tab at the top, find the file, and **Submit**.

Due:

Section A: Monday, October 10, 2016, 9:30 a.m. beginning of class

Section B: Tuesday, October 11 2016, 2:50 p.m. beginning of class

Resubmission **due**: Two weeks from the day the project is returned in class (*beginning of class*).