

# EECS 10: Assignment 3

Prepared by: Guantao Liu, Prof. Rainer Dömer

August 15, 2016

Due Monday August 22, 2016 at 11:00pm

## 1 Homework Problem 1: Paying off Credit Card Debt [20 points]

Given a credit limit, current balance, annual percentage rate (APR) and monthly payment, write a C program that computes and prints the interest paid and the remaining balance at the end of each monthly cycle. The program should continue printing the monthly values until the debt is paid off (i.e. the remaining balance becomes zero).

Your program should ask the user for the credit limit, balance on the card, APR (in percent) and the monthly payment as inputs in the beginning.

For example, if credit limit is \$5000, balance is \$3000, APR is 16.99% (floating point value 16.99), and monthly payment is \$300, then your program output should look as follows:

```
Enter the credit limit      :5000
Enter the balacne of the card :3000
Enter the APR               :16.99
Enter the monthly payment   :300
```

| Month | Balance | Interest | Payment | New Balance |
|-------|---------|----------|---------|-------------|
| 1     | 3000.00 | 42.47    | 300.00  | 2742.48     |
| 2     | 2742.48 | 38.83    | 300.00  | 2481.30     |
| 3     | 2481.30 | 35.13    | 300.00  | 2216.44     |
| 4     | 2216.44 | 31.38    | 300.00  | 1947.82     |
| 5     | 1947.82 | 27.58    | 300.00  | 1675.39     |
| 6     | 1675.39 | 23.72    | 300.00  | 1399.11     |
| 7     | 1399.11 | 19.81    | 300.00  | 1118.92     |
| 8     | 1118.92 | 15.84    | 300.00  | 834.77      |
| 9     | 834.77  | 11.82    | 300.00  | 546.58      |
| 10    | 546.58  | 7.74     | 300.00  | 254.32      |
| 11    | 254.32  | 3.60     | 257.92  | 0.00        |

It will take \$3257.92 over 11 month to pay off this debt

**Note:** For all dollar amounts and the APR value, print out exactly 2 digits after the decimal point. Also ensure that all the numbers in the output table line up nicely so that the decimal points are all at the same column position.

The first column "Month" keeps count of the number of months as the remaining debt diminishes each monthly cycle.

The second column "Balance" is the balance on the credit card at the beginning of each monthly cycle. For the first month, the balance is the value input by the user at the beginning of the program. For subsequent months, the balance is calculated using the following formula:

Balance = "New Balance" from the previous month.

The third column "Interest" is the interest accrued on the "Balance" at the end of each monthly cycle. It is calculated using the formula:

$$\text{Interest} = \text{Balance} * (\text{APR} / 100) / 12.$$

The fourth column is the "Payment" at the end of each monthly cycle. In our program, this will be a constant value that the user inputs at the beginning. In our example, this is \$300.

The fifth column is the "New Balance" at the end of each monthly cycle. It is calculated using the following formula:

$$\text{New Balance} = \text{Balance} + \text{Interest} - \text{Payment}.$$

You should submit your program code as file `creditcard.c`, a text file `creditcard.txt` briefly explaining how you designed your program, and a typescript `creditcard.script` which shows that you compile your program and run it. Use the following inputs to test your program:

Credit limit: \$5000  
Balance: \$3000  
APR: 16.99  
Payment: \$300

and

Credit limit: \$5000  
Balance: \$3000  
APR: 19.99  
Payment: \$300

## 2 Homework Problem 2: Collatz conjecture [30 points]

The Collatz conjecture is a mathematical problem that still is not solved today. Please see a nice introduction of the problem and its missing mathematical proof on Wikipedia:

[https://en.wikipedia.org/wiki/Collatz\\_conjecture](https://en.wikipedia.org/wiki/Collatz_conjecture)

Rather than solving the problem by proper mathematical proof, we want to write a C program that calculates the Collatz sequence for any number input by the user.

Specifically, the program will ask the user to input a positive number which will serve as the initial value of the Collatz sequence ( $a_0$ ). The program will then perform the Collatz function step by step until the terminating value 1 is reached.

For each step, the program should print the current value  $a_i$  on the screen. After the final value 1 is reached, the program should also print the number of steps ( $i$ ) needed.

As an example, the interface of the C program should look like the following:

```
=====  
Collatz Conjecture Test:  
=====
```

```
Enter a positive integer (or 0 to quit): 5  
Collatz sequence: 5, 16, 8, 4, 2, 1  
Collatz conjecture is true for n=5.  
Collatz sequence stops after 5 steps.
```

```
Enter a positive integer (or 0 to quit): 6  
Collatz sequence: 6, 3, 10, 5, 16, 8, 4, 2, 1  
Collatz conjecture is true for n=6.  
Collatz sequence stops after 8 steps.
```

```
Enter a positive integer (or 0 to quit): 7  
Collatz sequence: 7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1  
Collatz conjecture is true for n=7.  
Collatz sequence stops after 16 steps.
```

```
Enter a positive integer (or 0 to quit): 0  
Quit.
```

Since the Collatz conjecture is not proven, we need to ensure our program terminates even if the Collatz sequence would be infinite. To satisfy this condition for our algorithm, we will impose a maximum of 1 million steps. Then, if the sequence still has not terminated, the program should exit with the following message:

```
Stopping after 1 million steps.  
Collatz conjecture not clear for n=XXX.
```

You should submit your program code as file `collatz.c`, a text file `collatz.txt` briefly explaining how you designed your program, and a typescript `collatz.script` which shows that you successfully compile and run your program.

In your type script, please run your program for the initial values 6, 19, and 27.

### 3 Submission

Submission for the files will be similar to last week's assignment. Create a directory called `hw3/`. Put all the files for assignment 3 in that directory and run the `turnin10` command in the parent directory of `hw3/` to submit your homework.

### 4 Bonus [5 points]

Extend the Collatz conjecture program so that the maximum value reached in the sequence is printed, together with its corresponding step number.

For the example of starting value 27, the Collatz program should print the following result:

```
Collatz conjecture is true for n=27.  
Collatz sequence stops after 111 steps.  
Maximum value is 9232 at step 77.
```

To submit, use the same files as in Part 2, i.e. `collatz.c`, `collatz.txt`, and `collatz.script`.