CSCI 1133, Spring 2015
Programming Assignment 2

Assigned: Friday Feb 6, 2015
Due: 11:55pm Thursday Feb 12, 2015

This is ***not*** a collaborative assignment; you must design, implement and test the solution(s) on your own. You may not consult or work with anyone other than the course instructor or TAs. In addition, you may not include solutions or portions of solutions obtained from any source other than those provided in class. Obtaining or *providing* solutions to any homework problems for this class is considered academic misconduct. If you are not sure what this means, consult the class syllabus or discuss it with the course instructor.

This assignment requires writing two *separate* Python scripts that must be submitted online *prior* to the due date/time. Late submissions will not be accepted. Name each source code: `hw2`*prob*`.py` where *prob* is replaced by the corresponding problem letter: e.g., `hw2A.py` , `hw2B.py` Submit each source code file separately using the appropriate homework submission link on the Moodle website.

The total point value for each problem will be awarded for a solution that is *complete*, *correct*, and *well constructed*. A *"well constructed"* program entails good design, appropriate comments and general readability (descriptive names for variables and procedures, appropriate use of blank space, etc.). The following will result in a score reduction equal to a percentage of the total possible points:

- Incorrectly named/submitted source file (10%)
- Constraints not followed (40%)
- Failure to execute due to syntax errors (30%)

Note that your work will be graded using, and must function correctly with, the current version of Python 3 on CSE Labs UNIX machines. If you complete your programming assignment using a different system, it is your responsibility to ensure your programs work on CSELabs machines *prior* to submitting them.

A. (20 *points*) **Revolving Credit**[1]

Write a Python program that will input a credit-card account balance, then compute and output the following:

- Monthly interest charge
- Total amount due
- Minimum payment.

Monthly interest is charged based on the current account balance according to the following rate schedule: 1.5% of the first $1,000.00 and 1% of any amount over that. The total amount due is the current balance plus the computed monthly interest charge. The minimum payment is the total amount due if it is less than $10.00; otherwise it is either $10.00 or 10% of the total amount due (whichever is larger).

Your program should input the current account balance as a floating-point value, and then determine and output the Monthly interest, amount due and minimum payment. If the account balance is negative, then output "Invalid account balance" instead.

Constraints:

- Do not import or use any module functions
- Use floating point objects for all numeric quantities

Examples:

```
Enter current account balance: -16.25
Invalid account balance

Enter current account balance: 1355.26
Interest: 18.55
Amount Due: 1373.81
Minimum Payment: 137.38
```

```
Enter current account balance: 127.21
Interest: 1.91
Amount Due: 129.12
Minimum Payment: 12.91

Enter current account balance: 56.23
Interest: 0.84
Amount Due: 57.07
Minimum Payment: 10.00

Enter current account balance: 8.96
Interest: 0.13
Amount Due: 9.09
Minimum Payment: 9.09
```

---

[1] adapted from Walter Savitch, *Problem Solving with C++, 9th edition*

B. (20 *points*) **Lexicographic Ordering**

Write a Python program that will input three separate string values and output them in increasing *lexicographic* order. Recall that *lexicographic* order is similar to *alphabetic* order but includes all alphanumeric symbols and is case-sensitive).

Constraints:

- Do not import or use any module functions
- Use only simple comparison and selection operations, do not use any object methods or list structures

Examples:

```
Enter first string: dog
Enter second string: cat
Enter third string: dolphin
cat
dog
dolphin

Enter first string: dog
Enter second string: cat
Enter third string: Dolphin
Dolphin
cat
dog
```