13 Feb 2020

COMP281 2020 – Assignment 1

- In the following, you will find the problems that constitute Assignment 1. They will be also available on the *online judging (OJ)* system available at https://student.csc.liv.ac.uk/JudgeOnline

- You need to write a C program (not C++ or C#) that solves each problem – it must read the input, as specified in the problem description then print the solution to the given problem for that input.
  - Note that code is "correct" only if it **correctly implements a solution to the problem stated** in the assignment, not "if online judge accepts it".
  - That is, even if OJ accepts your code, it could be wrong. Read the problems carefully.

- Input is read from the standard input, in the same way that you read input from the keyboard as shown in lectures (e.g., using scanf). Output is also printed to the standard output, as you have seen (e.g., using printf).

- When you are satisfied that your C programs work correctly, you must submit them through the departmental submission system.
  - Even if the program is not correct, still submit whatever you have! You can still earn points if certain parts of the code are done right.

- You must also include a brief report describing your solutions to the problems. This should be maximum two sides of A4 paper and should give a description of how each of your solutions works. This should include describing the algorithm used to reach the solution, describing your use of any C language features (that were not discussed in lectures) and identifying any resources that you have used to help you solve the problems.

- This assignment is worth 50% of the total mark for COMP281.
  - All five problems in this assignment are weighted equally.
  - For each problem, you can earn a total of 20 points
    - 10 points for "Functionality and Correctness" awarded for programs that **correctly** solve the problem for all test cases.
    - 8 points for "Programming style, use of comments, indentation and identifiers" awarded depending on the style, comments and efficiency of the solution
    - 2 points for the quality and depth of the accompanying report
  - The final grade results from normalising the earned points to a scale of 100.
  - See separate "comp281-detailed-marking-guidelines.pdf" for more details.

Submission Instructions

- Create a folder, and name it using your Student ID and User ID, e.g. 201234567_ sgpj6
- In the folder, there should be 6 files:
    - 1 report file, in PDF format. Name it with your Student ID and User ID, e.g. 201234567_ sgpj6.pdf
    - 5 source code files. Name each using the Problem Number, e.g. 1006.c
        - In your source code, include your Student Info and Problem Info, e.g.:

```
/*
 * Student ID: 201234567
 * Student Name: Phil Jimmieson
 * Email: phil.jimmieson@student.liverpool.ac.uk
 *
 * User: sgpj6
 *
 * Problem ID: 1006
 * RunID: 22456
 * Result: Accepted
 */
```

        - The OJ provides a RunID, which is different from the Problem ID.
        - The Result is one of the following: Accepted, Wrong Answer, Presentation Error, Time Limit Exceeded, Memory Limit Exceeded, Output Limit Exceeded, Runtime Error, Compile Error.
- Compress the folder into a single zip file, and name it as, e.g. 201234567_ sgpj6.zip
    - Use the standard (pkzip) zip file format:
      https://en.wikipedia.org/wiki/Zip_%28file_format%29
      which is supported by winzip, winrar, etc. on Windows/Mac OS X, and 'zip' on Linux
    - Test your zip file before submitting.
- Submit this zip file using the departmental submission system at
  http://www.csc.liv.ac.uk/cgi-bin/submit.pl
  Only the file submitted through this link will be marked.
- The **deadline** for this assignment submission is **24-Feb-2020 17:00**
- Penalties for late submission apply in accordance with departmental policy as set out in the student handbook, which can be found at: http://intranet.csc.liv.ac.uk/student/ug-handbook.pdf

1. Problem 1013

Title: Record marks

Description

You are in charge of recording marks for a group of students. Input a list of marks. Input ends with 0 (0 itself is not someone's mark). Output the number of students who scored 1) greater than or equal to 85; 2) between 60 and 84; 3) strictly less than 60.

Make sure you use good coding standards: i.e., create separate functions for processing the input and writing the output, and be consistent in the way you name variables (also see the “COMP 281 Detailed Marking Guidelines”).

Input

Output

Sample Input

88 71 68 70 59 81 91 42 66 77 83 0

Sample Output

>=85:2
60-84:7
<60:2

HINT

Remember that there are different ways of obtaining input. For this assignment "scanf" might be convenient, but make sure to use the correct syntax (it requires a 'pointer' as a second argument.)

2. Problem 1014

Title: Area and circumference of circles

Description

In this exercise you have to compute the area and circumference of a series of circles and output their sum. Specifically, the program will take the radius of two circles as input (r1 <= r2, both integers) and will output the sum of the areas and the circumferences of all circles starting with r1 and increasing at each step the radius by '1' until radius r2 has been reached. As an example, suppose 'r2-r1 = 2' then the program has to compute the sum of the areas and circumferences of three circles with radii r1, r1+1,r2.

Remember that the area of a circle equals Pi*r^2 and the circumference equals 2Pi*r.

Set Pi to 3.14

Input

Two integers r1 and r2 with r1<=r2.

Output

Two floats, sumofareas and sumofcircumferences.

The result should be to 3 digits precision.

Sample Input

3 4

Sample Output

78.500
43.960

3. Problem 1015

Title: ASCII code

Description

Convert integers (ASCII codes) to characters.

Input

A list of positive integers separated by whitespaces (spaces, newlines, TABs). The input ends with EOF.

Output

A list of characters. (Do not add end-of-line at the end of the output.)

Sample Input

```
72 101 108 108 111 44
32 119 111 114 108 100 33
```

Sample Output

```
Hello, world!
```

HINT

H's ASCII code is 72. Blank space's ASCII code is 32.

4. Problem 1025

Title: Largest common factor and smallest common multiple

Description

Input two positive integers. Compute their largest common factor and smallest common multiple.

Input

Output

Sample Input

2 3

Sample Output

1 6

5. Problem 1030

Title: Precise division

Description

8/13=0.615384615384615384615384...

For 8/13, the 5-th digit after the decimal point is 8.

Given three positive integers a, b, and n (all at most 60000), you are asked to compute a/b and print out the n-th digit after the decimal point.

Input

a b n

Output

The n-th digit after the decimal point of a/b.

Sample Input

8 13 5

Sample Output

8