

COMP281 Assignment 2 – C Programming

- The following 5 problems will be available as Assignment 2 on the online judging system available at <http://intranet.csc.liv.ac.uk/JudgeOnline/>
- You need to write a valid C program that solves each of these problems – it must read the input, as specified in the problem description then print the solution to the given problem for that input.
- 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 programs work correctly, you must submit them through the departmental submission system, as described below.
- You must also include a brief report describing your solutions to the problems. This should be up to one side of A4 paper (although there is no penalty for a longer report) 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), identifying any resources that you have used to help you solve the problems and describing any joint work you may have done with other students.
- This assignment is worth 30% of the total mark for COMP281
 - 50% of the marks will be awarded for programs that correctly solve the problem for all test cases. All problems are weighted equally and each counts for 20% of this total.
 - 40% of the marks will be awarded depending on the style, comments and efficiency of the solution. All problems are weighted equally and each counts for 20% of this total.
 - 10% of the marks will be awarded for the quality and depth of the accompanying report
 - See separate marking guidelines for more details.

Submission Instructions

- Name each of your c files according to the problem number; i.e., 1029.c, 1032.c, 1041.c, 1043.c and 1044.c. Place all five of your C files and your report (in .pdf format) into a single zip file.
- All questions have also been/will also be posted on www.csc.liv.ac.uk/JudgeOnline2
- Before you submit your solutions, please first test them using the online judge. **You are required to** include the “Result” and the “RunID” in your C code as comments. **The OJ provides a RunID. RunIDs are not problem IDs.**
 - Example: the problem is 10xx . The solution has been accepted by the OJ, and the runID is 2033. You add to your code: `/* 2033 Accepted */`
 - Result is one of the following: Accepted, Wrong Answer, Presentation Error, Time Limit Exceeded, Memory Limit Exceeded, Output Limit Exceeded, Runtime Error, Compile Error
- Submit this zip file using the departmental submission system at :
<http://www.csc.liv.ac.uk/cgi-bin/submit.pl>

The deadline for this assignment is Wednesday March 4th, 4:00pm

- Penalties for late submission apply in accordance with departmental policy as set out in the student handbook, which can be found at:
<http://www.csc.liv.ac.uk/student/ugpdfhandbook.pdf>

Problem 1029

Title: ROT R 3

Description

ROT R 3 ("Rotate Right by 3") is a simple substitution cipher to encode letters. Applying ROT R 3 to a piece of text merely requires examining its alphabetic characters and replacing each one by the letter 3 places further or 'right' of it along in the alphabet, wrapping back to the beginning if necessary. A becomes D, B becomes E, and so on up to W, which becomes Z, then the sequence continues at the beginning of the alphabet: X becomes A, Y becomes B, and Z, becomes C. Only those letters, which occur in the English alphabet, are affected; numbers, symbols, whitespace, and all other characters are left unchanged.

Input

One line of characters (ends with EOF)

Output

The result after applying ROT R 3 to the input

Sample Input

ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz!

Sample Output

DEFGHIJKLMNOPQRSTUVWXYZABCdefghijklmnopqrstuvwxyzabc!

Problem 1032

Title: String search

Description

Input three strings s1, s2, and s3 (one for each line). They may contain whitespaces. Output the number of times s3 and s2 occur as a substring of s1.

For example, if the input strings are "hello world", "europe" and "wor", then the output should be "0 1". As another example, if the input strings are "hello world hello", "hello world" and "hel", then the output should be "1 2". If the input strings are "hello world", "hello" and "helo", then the output should be "1 0".

You are **not** allowed to use system functions defined in string.h.

Input

s1, s2 and s3.

Output

Two numbers.

Sample Input

```
Hello World! Hello
Hello
World
```

Sample Output

```
2 1
```

Problem 1041

Title: Dot Product of two matrices

Description

Input three positive integers n , m , and p . Then input two matrices. The first one is a n (rows) by m (columns) matrix. The second one is a m (rows) by p (columns) matrix. Output the matrix product (should be n rows by p columns). You are required to store the matrices in heap memory.

Input

n , m , p , and two matrices

Output

Product of these two matrices

Sample Input

```
4
3
2
14 9 3
2 11 15
0 12 17
5 2 3
12 25
9 10
8 5
```

Sample Output

```
273 455
243 235
244 205
102 160
```

Problem 1043

Title: Sort Strings

Description

Input n and then followed by n strings, one string per line. Subsequently this is followed by input m , and then followed by m strings (one per line). Strings consist of lowercase English letters only. The maximum

string length is 100 (at most 99 letters, plus '\0'). However, most strings are very short. That is, you should dynamically allocate just enough memory to store each string. You should also dynamically allocate just enough space for storing n+m string pointers.

Please sort these n+m strings based on their lengths in descending order. String lengths are unique. There will be no ties. Output the sorted list of strings.

Input

n and then followed by n strings. m and then followed by m strings.

Output

Sorted list, one string per line

Sample Input

```
5
abcdefg h i j k
abcde
a
abcdefghijklmnopabcdefghijklmnop
zzzzzzzzzzzzzzzzzzzzzzzzzzzzzz
3
bb
ccc
hhhhhhhhhhhhhhhh
```

Sample Output

```
abcdefghijklmnopabcdefghijklmnop
zzzzzzzzzzzzzzzzzzzzzzzzzzzzzz
hhhhhhhhhhhhhhhhhh
abcdefg h i j k
abcde
ccc
bb
a
```

Problem 1044

Title: 2 raised to the power of x

Description

2^x stands for 2 raised to the power of x. $2^4=16$ and the sum of its digits is $1+6=7$. $2^5=32$ and the sum of its digits is $3+2=5$. Input x. x is a nonnegative integer that is at most 10000. Output the sum of the digits of 2^x .

Hint: Problem 1036 is a good start.

Input

x

Output

result (one integer)

Sample Input

4

Sample Output

7