INDIAN STATISTICAL INSTITUTE

MTech(CS) I year 2020-2021

Subject: Computing Laboratory

Surprise Test (25 January, 2021)

Total: $5 \times 20 = 100$ marks Duration: 3 hours

SUBMISSION INSTRUCTIONS

1. Please make sure that your programs adhere strictly to the specified input and output format.
2. Please adhere to the file naming conventions discussed in class.

   IMPORTANT: Insert a single alpha-numeric string of your choice, 6-8 characters long, in the name given above as shown in the examples below. Think of this string as something like a security password, except that you are not required to remember the string.

   Examples: `cs20XX-surprise-abcdef-prog1.c`, `mnopqr-cs20XX-surprise-prog2.c`, `cs20XX-surprise-prog2-uvwxyz.c`.
3. Please upload your programs to `/user1/perm/pdslab/2020/surprise-test/` on the ISI server (192.168.64.35), as advised in class.

NOTE: Write the programs in C/C++. Unless otherwise specified, all programs should take the required input from stdin, and print the desired output to stdout.

Q1. Let us define the `STRETCH` operation on a string that increases the number of occurrences of its characters retaining their ordering in the original string. E.g., if a string "computing" is stretched by a factor of 2 then it will turn into "ccoommppuuttiinngg". Write a program that will take a string and a stretching factor as inputs and show the stretched string as the output.

Input Format

Inputs will be provided as command-line arguments in the following format. The first argument will be the string and the second argument will be the stretching factor.

Output Format

Output is to be printed on the standard output.

Command-line Arguments

```
./prog1 <string> <stretching_factor>
```

Sample Input 0

```
./prog1 lab 3
```

Sample Output 0

```
lllaaabbb
```

Sample Input 1

```
./prog1 balloon 2
```

Sample Output 1

```
bbaalllloooonn
```

Q2. A number is `PSEUDOPERFECT` if the sum of all or some of its proper divisors is equal to the number itself. Write a program to verify whether a given number is pseudoperfect or not.

Input Format

Input number will be provided from stdin.

Output Format

Output is to be printed on the standard output. It must show `PSEUDOPERFECT` or `NOT PSEUDOPERFECT` based on whether the number is pseudoperfect or not. E.g., 12 is a pseudoperfect number because from its divisors 1, 2, 3, 4 and 6, we can ignore 4 and derive $1 + 2 + 3 + 6 = 12$.

Sample Input 0

```
20
```

Sample Output 0

```
PSEUDOPERFECT
```

Sample Input 1

```
49
```

Sample Output 1

```
NOT PSEUDOPERFECT
```

Q3. Suppose you have a set of contact details comprising names along with mobile numbers. Given a particular search string, may be a portion of a name or a number, you need to return all the corresponding details. Write a program that will list up all predicted contact details for the given search string in the alphabetical order of contact names.

Input Format

Input will be provided in two different forms. The contact details will be provided in a separate file (name to be taken from command-line arguments) in the following format.

The first line of input will contain n, where n is the number of contacts. This will be followed by n more lines each containing the contact names and numbers separated by spaces, not necessarily in alphabetical order.

The alphanumeric search string will also be provided as a command-line argument alongside the filename.

Output Format

Output is to be printed on the standard output. Output will print the list of all predicted contact names (case-insensitive match) and numbers, separated by spaces, for the given search string in alphabetical order of contact names.

Command-line Arguments

```
./prog3 <input_filename> <search_string>
```

Sample Input 0

```
Contents of input.txt:
3
Ansuman Banerjee 9000000001
Mandar Mitra 9000000002
Malay Bhattacharyya 9000000003

Input from command-line:
./prog3 input.txt man
```

Sample Output 0

Output on the terminal:

```
Ansuman Banerjee 9000000001
Mandar Mitra 9000000002
```

Sample Input 1

```
Contents of input.txt:
3
Emergency 9876543210
Workshop 9876543299
Library 9976543210

Input from command-line:
./prog3 input.txt 99
```

Sample Output 1

Output on the terminal:

```
Library 9976543210
Workshop 9876543299
```

Q4. Without using the getpass() function, write a program to take a password from the user and verify its strength. Mask the password text with character '?'. Strength of the password is determined as follows.

- If the password contains at least one lowercase alphabet, one uppercase alphabet, one digit, and one special character, then return STRONG.
- Otherwise, return WEAK.

Input Format

Input is to be taken from the standard input.

Output Format

Output is to be printed on the standard output.

Sample Input 0

```
CLab@21 (to be shown as ??????? on the screen)
```

Sample Output 0

```
STRONG
```

Sample Input 0

```
Python### (to be shown as ????????? on the screen)
```

Sample Output 0

```
WEAK
```

Q5. Let us define a sequence $a_0, a_1, a_2, \ldots, a_{n-1}$ as Λ-*bitonic* if there exists a j, $0 \leq j < n$, such that $a_0 < a_1 < \cdots < a_j > a_{j+1} > \cdots > a_{n-1}$. Consider an $m \times n$ matrix A consisting of integer entries, such that each row and each column of the matrix forms a Λ-bitonic sequence. Write a program to efficiently find the largest element of the matrix.

Input Format

Input will be provided via standard input in the following format. The first line of input consists of a pair of integers, namely the number of rows (m) and number of columns (n) of the matrix A. It follows by the elements of A, providing each row in a single line following their order in the matrix.

Output Format

Output is to be printed on the standard output in the following format. The output will print the largest element of the input matrix A.

Sample Input 0

```
1 8
-10 -5 0 5 10 15 20 10
```

Sample Output 0

```
20
```

Sample Input 1

```
3 6
10 19 30 28 26 25
15 25 35 50 41 22
10 20 22 38 40 20
```

Sample Output 1

```
50
```