

Quiz 1 – var 4 - (5 points) – 40min

#1 (String) – 1.5points

Hangman

Let's say the word the player has to guess is "EVAPORATE". For this exercise, write the logic that asks a player to guess a letter and displays letters in the clue word that were guessed correctly. For now, let the player guess an infinite number of times until they get the entire word. As a bonus, keep track of the letters the player guessed and display a different message if the player tries to guess that letter again. Remember to stop the game when all the letters have been guessed correctly! Don't worry about choosing a word randomly or keeping track of the number of guesses the player has remaining - we will deal with those in a future exercise.

An example interaction can look like this:

```
>>> Welcome to Hangman!
```

```
-----
```

```
>>> Guess your letter: S
```

```
Incorrect!
```

```
>>> Guess your letter: E
```

```
E _ _ _ _ _ E
```

```
...
```

And so on, until the player gets the word.

#2 (Dict) – 1 points

Replace Letters With Position In Alphabet

Create a program that takes a string and replaces each letter with its appropriate position in the alphabet.

Rules:

Replace all letters with position in alphabet.

If anything in the string isn't a letter, ignore it.

a is 1, b is 2, c is 3, etc, etc.

Examples:

"We have a lot of rain in June." → "23 5 8 1 22 5 1 12 15 20 15 6 18 1 9 14 9 14 10 21 14 5"

"The river stole the gods." → "20 8 5 18 9 22 5 18 19 20 15 12 5 20 8 5 7 15 4 19"

"Wow, does that work?" → "23 15 23 4 15 5 19 20 8 1 20 23 15 18 11"

#3 (math) – 1 point

Here is a variation of another popular task for practicing dynamic programming approach (though of course precise algorithm is not explained by these words).

11 5 3 17 2 13 19 7

The Rabbit is going to cross the river. There is a straight chain of tiny isles across the flow and the animal should jump from one to another because it surely could not swim.

At each of the isles there are some candies. When the Rabbit arrives to the new isle, it collects all the candies here.

However, the Rabbit could not jump directly to the next isle in the chain - it just is too strong to make short jumps. So, instead, it can jump over the one or two isles (i.e. from the 1st for example to 3rd or 4th but not to 2nd or 5th and further). Also the Rabbit could not jump back.

You can see the sample of the Rabbit's path on the drawing above. It visits 1st, 3rd, 6th and 8th isles and collects:

$$11 + 3 + 13 + 7 = 34$$

the amount of 34 candies. Obviously he could do better if the path is chosen more wisely.

Your task is to choose the best path for Rabbit over the given chain of isles - i.e. to maximize the amount of the candies collected. Note that Rabbit starts from 1st isle and finishes either on the Nth or (N-1)th where N is the total number of isles (because from these two it will necessarily jump immediately to the other bank).

Input data will contain the number of test-cases in the first line.

Next lines contain one test-case each - i.e. one chain of isles, described by the array of numbers - amounts of candies at each isle.

Answer should contain the maximum possible amount of candies gathered for each test case.

Example:

input data:

2

11 5 3 17 2 13 19 7

9 7 12 7 16 3 7 17 14 13 4 6 11 6 3 3 5 4 11 3 15 12 14 2 15 19 11 12

answer:

48 157

#4 (math) – 0.75 points

When we speak about arithmetic progression (or arithmetic sequence) we mean a series of numbers with a special property - each value is followed by the other, greater by predefined amount (step).

I.e. difference of $(K+1)$ -th and K -th values is a constant. Here are examples of sequences

1 2 3 4 5 6 7 ...

4 6 8 10 12 14 16...

10 13 16 19 22 25 28...

Since so, arithmetic sequence is completely defined by the first member (A) and the increment value - step size - (B). First few members could be expressed as $A + (A + B) + (A + 2B) + (A + 3B) + \dots$

Input data: first line contains the number of test-cases.

Other lines contain test-cases in form of triplets of values $A B N$ where A is the first value of the sequence, B is the step size and N is the number of first values which should be accounted.

Answer: you are to output results (sums of N first members) for each sequence, separated by spaces.

Example:

data:

2

5 2 3

3 0 10

answer: 21 30

#5 (Input/Output from a file) – 0.5 points

Write a python program to find the longest words from a file. The longest one must be written into the file.

#6 (math) – 0.25 points

Pythagorean Theorem

In this task you will use this theorem to write a program which could determine, whether the triangle is right, or it is acute, or obtuse:

for acute triangle its longest side is shorter than hypotenuse should be;

for obtuse triangle its longest side is longer than hypotenuse should be.

Input data contains the number of triangles in the first line.

Next lines describe one triangle each. Descriptions consist of three values - lengths of sides. Largest value would always be the last of three for simplicity.

Answers should have one of the letters R (right), A (acute) or O (obtuse) for each of triangles. Letters should be separated by spaces.

Example:

input data:

3

6 8 9

9 12 15

16 12 22

answer: A R O