Quiz 1 var 3 - (5 points) – 40min

#1 (Clock) – 1.5 points

The face of analog clock has two hands and is proportionally divided by 12 marks. The shorter hour hand makes the whole turn (360 degrees) in 12 hours, while the longer minute hand makes the whole circle each hour. See Clock Face on wiki for more details.
Suppose, the Cartesian Coordinate System (i.e. ordinary rectangular coordinate grid) is placed upon the clock face so that the center of the face has coordinates 10, 10 and Y axis is directed upwards while X axis is directed to the right (i.e. at 3:00 minute hand is parallel to Y axis and hour hand is parallel to X axis).
Assuming the length of the minute hand be 9 and the length of the hour hand be 6 you are to find coordinates of the hand ends for each given time - e.g. (16 10) and (10 19) for the time 3:00.
Input data contain the number of test cases.
Following line contains the test cases themselves in form 03:15, 21:44 etc.
Answer should contain four real numbers for each test case - X and Y coordinates for hour hand, then X and Y coordinate for minute hand. All values should be simply separated with spaces.

#2 (math) – 1 points

Suppose we select some initial number X and then build the sequence of values by the following rules:
if X is even (i.e. X modulo 2 = 0) then Xnext = X / 2
else Xnext = 3 * X + 1
I.e. if X is odd, sequence grows - and if it is even, sequence decreases. For example, with X = 15 we have sequence:
15 46 23 70 35 106 53 160 80 40 20 10 5 16 8 4 2 1
After the sequence reaches 1 it enters the loop 1 4 2 1 4 2 1....
The intrigue is in the fact that any starting number X gives the sequence which sooner or later reaches 1 - however though this Collatz Conjecture was expressed in 1937, up to now no one could find a proof that it is really so for any X or could not find a counterexample (i.e. number for which sequence did not end with 1 - either entering some bigger loop or growing infinitely).
Your task is for given numbers to calculate how many steps are necessary to come to 1.
Input data contains number of test-cases in the first line.
Second line contains the test-cases - i.e. the values for which calculations should be performed.
Answer should contain the same amount of results, each of them being the count of steps for getting Collatz Sequence to 1.
For example:
input data:
3
2 15 97
answer:
1 17 118

#3 (Input/Output file) – 1 points

Write a Python program to combine each line from first file with the corresponding line in second file. And the output should be written into another third file.

#4 (Math) – 0.75 points

So let us create the array (or list) of prime numbers in ascending order, i.e.:
[2, 3, 5, 7, 11, 13, 17, 19, 23, 29, ...]
And then print the primes corresponding to the indices given in the input data.
Input data will contain the amount of primes to print in the first line.
Next line will contain indices of array of primes for which values should be printed. They will be in range from 1 to 200000.
Answer should contain prime numbers corresponding the specified positions of the array.
Note that for this task we start indexing an array from 1 rather than 0 (this may help you in checking your program with many lists of primes which could be found online).
Example:
input data:

```
4
7 1 199999 4
```

answer:

```
17 2 2750131 7
```

#5 (List) – 0.5 points

Group Monotonicity
Create a program which returns the indices where the monotonicity of a 1-D array changes. If there are none, return an empty list. A monotonic list is one that is either non-increasing or non-decreasing.
Examples

[0, 1] ➞ []

[0, 2, 1] ➞ [1]

[0, 1, 1, 0] ➞ [2]

#6 (math) – 0.25 points

Being able to calculate area of triangle is quite important since many more complex tasks are often easily reduced to this (and we will use it too later).
One of the oldest known methods is Heron's Formula which takes as inputs the lengths of the triangle's sides.
In this problem you however is to write a program which uses X and Y coordinates of the triangle's vertices instead. So you can use either this formula somehow or find another one.
Input data will contain the number of triangles to process.
Next lines will contain 6 values each, in order X1 Y1 X2 Y2 X3 Y3, describing three vertices of a triangle.
Answer should give areas of triangles separated by space (precision about 1e-7 is expected).
Example:
data:

```
3
1 3 9 5 6 0
1 0 0 1 10000 10000
7886 5954 9953 2425 6250 2108
```

answer: 17 9999.5 6861563