Instructions

- Send in your solutions in a zip file within 24 hrs to postursol@greyorangerobotics.com
- For any query, call: Srijan Choudhary (+91 8130824397) / Akash Gupta (+91 9560323262).
- You can give comments in the code to explain.

Problem Statements

1. Design a system and write basic code for an online exam server for an exam like GRE in which question difficulty increases if you answer correctly and decreases if you answer wrong.

Discuss the data structure you will use to store the questions, algorithms to fetch the question by its difficulty, etc. Try to be modular and well-designed.

While writing code you can assume all the data is written to memory and you need not store it to disk. You can use any language you want.

Notes:

- Questions are multiple choice questions
- Difficulty scale of question is 1-10, 1 being the easiest and 10 being the hardest.
- If two continuous questions are answered wrong, decrease the difficulty by 1 and if two questions answered right increase the difficulty by 1.
- Test starts with a question of difficulty level of 4.
- Question carries marks equal to its difficulty.

---

2. Write a program to count the occurrence of each word in a given text. Assume that input is taken from a file named “**input.txt**”. Write your output to a file “**output.txt**”, and it should be a list of all words, with their respective counts.

Try to optimize space and time complexities. Make assumptions if required, and mention them clearly in comments. Also, try to structure your code using OOP concepts.

Sample input.txt and output.txt are provided with this question sheet.

---

3. You are given the price of a particular stock for **N** consecutive days. If you were only permitted to buy one share of the stock and sell one share of the stock, design an algorithm to find the best times to buy and sell.

The input will have N+1 lines, with the 1st line giving the value of N (N<=100000), and the next N lines containing a positive integer, each giving the value of the stock at the corresponding day.

The output should have two integers – the buying day and the selling day.

Sample Input:

```
6
10
13
30
21
2
11
```

Sample Output:

```
1
3
```