Aequilibrium | Technical Assignment

For the answers to these questions, please direct us to the code in your Github repository.

Part 1: Aequilibrium | The Castle Company

Aequilibrium is in the business of building castles (we really aren't, but let's pretend). Now, we also believe in quality aesthetics, so we only want to build castles in two types of places:

a. Peaks

b. Valleys

Let's say you have an array of integers that describes a stretch of land, where each integer represents the current height of the land. Can you write a function that reads that array and returns the number of castles that Aequilibrium should build on that stretch of land? You can write this solution in whatever language you like and provide a way to test it.

You can make the following assumptions:

- You can always build a castle at the start of the array, provided it is non-empty
- We only want to build one castle per peak or valley.
- A peak is an integer or series of integers that is above the immediately preceding and following ints. For example, in the sequence [2,6,6,6,3] the sequence of 3 6s is a peak.
- A valley is an integer or series of integers that is below the immediately preceding and following ints. For example, in the sequence [6,1,4] the "1" would be a valley.

Part 2: Aequilibrium | The Transformation Company

Aequilibrium does love transforming… people, lives, teams, companies. And there's no better representation of transformation than Hasbro's *Transformers*, the classic television series featuring heroic Autobots raging their battle to destroy the evil forces of the Deceptions.

Please watch this video:
https://www.youtube.com/watch?v=nLS2N9mHWaw

Build a web application for the following:

The Transformers are at war and **you** are in charge of settling the score! You're to evaluate who wins a fight between the Autobots and the Decepticons. Here are the rules.

Each Transformer has the following criteria on their tech spec (see http://www.ntfa.net/ntfa/techspecs/index.php?cat=Gen1&group=DeceptPZ&char=Predaking for an example):

- Strength
- Intelligence
- Speed
- Endurance
- Rank
- Courage

- Firepower
- Skill

All of these criteria are ranked from 1 to 10.

The “overall rating” of a Transformer is the following formula:
(Strength + Intelligence + Speed + Endurance + Firepower)

Each Transformer must either be an Autobot or a Deception.

Your program should take input that describes a group of Transformers and based on that group displays:

a. The number of battles
b. The winning team
c. The surviving members of the losing team

The basic rules of the battle are:

- The teams should be sorted by rank and faced off one on one against each other in order to determine a victor, the loser is eliminated
- A battle between opponents uses the following rules:
    - If any fighter is down 4 or more points of courage and 3 or more points of strength compared to their opponent, the opponent automatically wins the face-off regardless of overall rating (opponent has ran away)
    - Otherwise, if one of the fighters is 3 or more points of skill above their opponent, they win the fight regardless of overall rating
    - The winner is the Transformer with the highest overall rating
- In the event of a tie, both Transformers are considered destroyed
- Any Transformers who don’t have a fight are skipped (i.e. if it’s a team of 2 vs. a team of 1, there’s only going to be one battle)
- The team who eliminated the largest number of the opposing team is the winner

Special rules:

- Any Transformer named Optimus Prime or Predaking wins his fight automatically regardless of any other criteria
- In the event either of the above face each other (or a duplicate of each other), the game immediately ends with all competitors destroyed

For example, given the following input:

Soundwave, D, 8,9,2,6,7,5,6,10
Bluestreak, A, 6,6,7,9,5,2,9,7
Hubcap: A, 4,4,4,4,4,4,4,4

The output should be:

1 battle
Winning team (Decepticons): Soundwave
Survivors from the losing team (Autobots): Hubcap