Programming test for Software Developers (at all levels)

Part I (30 minutes)

Write a function to calculate a score for a candidate's response for an essay type question based on how closely it matches the correct answer.

Assume that the function takes two string arguments:

- Candidate's response (Response)
- Correct Answer (CorrectAnswer)

The function needs to identify each word in the Candidate's response that matches with a word in the Correct Answer and assign points based on the following rule:

- Numbers - 4 points
- Words with more than 7 characters - 3 points
- Words with less than 5 characters - 0 points
- All other words - 1 point

Calculate "Maximum Possible Score" as Sum of points for each word in the Correct Response String

Calculate "Points Scored" as Sum of points for each word in the "Candidate Response" that has a **match with a word** in the Correct Answer string.

After calculating both "Maximum Possible Score" and "Points Scored", the function should return the percentage ratio of the "Points Score" and the "Maximum Possible Score"

Read through the following example to understand the required outcome of the function

Correct Answer:

"There are twenty-four hours in a day, 30 days in a month, and 12 months in the calendar year."
1 0 3 1 0 0 0 4 0 0 0 1 0 4 1 0 0 1 0

So the "Maximum Possible Score" (A) is 16.

Candidate Response:

"There are Twenty-Four hours in a day. A year has 14 months."
1 0 3 1 0 0 0 0 0 0 0 1

So the "Points Scored" (B) is 6. The percentage score (B/A %) is 37.5%.

- There is some ambiguity in the above requirement specification. It is expected that you will make necessary assumptions and note these down along with your solution as comments.
- Extra points for spotting any errors with the test data provided above

Part II (30 Minutes)

Write the following queries based on the provided database schema described below for a blog system:

- Number of posts in each category including categories with no posts.
- Average number of categories used within each blog

Additionally, if you are really good with SQL, you can attempt the query below:

- Select Top 3 posters (users) for each category

Database Schema

* You don't need to use a SQL Database for this test, simply review the table structure above and write down the sql queries in a text file.

User

- UserId
- UserName
- Password

Blog

- BlogId
- UserId
- BlogName

Post

- PostId
- BlogId
- PostName
- PostText

Category

- CategoryId
- CategoryName

PostCategory

- PostId
- CategoryId

Part III (30 minutes)

1. Write a function that determines if a string starts with an upper-case letters A-Z
2. Write a recursive function that reverses the order of a string
3. Write code to remove all items from a list that is contained in another list.