

IntervAI: Your Virtual Interview Assistant

Assessment Report



Name: ishan

Field: Artificial intelligence

Introduction: hello I am Yash Patni a passionate and Ambitious data scientist with 5 years experience driving business growth through data driven insights my specialties are machine learning artificial intelligence data visualisation and statistical modelling with the strong educational background in Computer Science I have developed a unique blend of Technical expertise in business acumen and business acumen achievements include developing predictive models that increase sales by 25%

Confidence: Confident

Polarity: 0.42261904761904756

Subjectivity: 0.6047619047619046

Efficiency: Needs Improvement

Aptitude Score: 100.0

Technical Score: 80.0

Average Coding Round Score: 18.0

Question 1: Write a function that takes an array of integers and returns the sum of its elements.

Code Submitted: asdasdas

Score: 0/100

Feedback Summary: The provided answer `asdasdas` is not a valid program in any programming language. It cannot be analyzed for correctness, efficiency, readability, or adherence to best practices.

Question 2: Write a function that takes a string and returns a new string with all vowels removed.

Code Submitted: dasdas

Score: 0/100

Feedback Summary: The provided answer is not a valid program and does not address the problem statement. It should return a string with all vowels removed.

Question 3: Write a function that takes two strings and returns the longest common substring.

Code Submitted: `def longest_common_substring(s1, s2):`

```
    # Get the lengths of both strings
```

```
    len1, len2 = len(s1), len(s2)
```

```
    # Initialize the DP table with zeros
```

```
    dp = [[0] * (len2 + 1) for _ in range(len1 + 1)]
```

```
    # Variables to track the maximum length and end index of the substring
```

```
    max_len = 0
```

```
end_idx = 0
```

```
# Fill the DP table
```

```
for i in range(1, len1 + 1):
```

```
    for j in range(1, len2 + 1):
```

```
        if s1[i - 1] == s2[j - 1]:
```

```
            dp[i][j] = dp[i - 1][j - 1] + 1
```

```
            # If we found a longer common substring, update the max length
```

```
            if dp[i][j] > max_len:
```

```
                max_len = dp[i][j]
```

```
                end_idx = i - 1
```

```
        else:
```

```
            dp[i][j] = 0
```

```
# The longest common substring is from end_idx - max_len + 1 to end_idx in s1
```

```
return s1[end_idx - max_len + 1:end_idx + 1]
```

Score: 90/100

Feedback Summary: The solution is correct and efficient. The code is clear and readable, and it adheres to best practices.

Question 4: Write a function that takes a binary tree and returns its maximum depth.

Code Submitted: sss

Score: 0/100

Feedback Summary: This answer shouldn't be answered like that. It should be written in a programming language. Please provide a coding answer to this question.

Question 5: Write a function that takes a linked list and returns the k-th element from the end.

Code Submitted: as

Score: 0/100

Feedback Summary: The answer is incomplete. It's not a valid program.