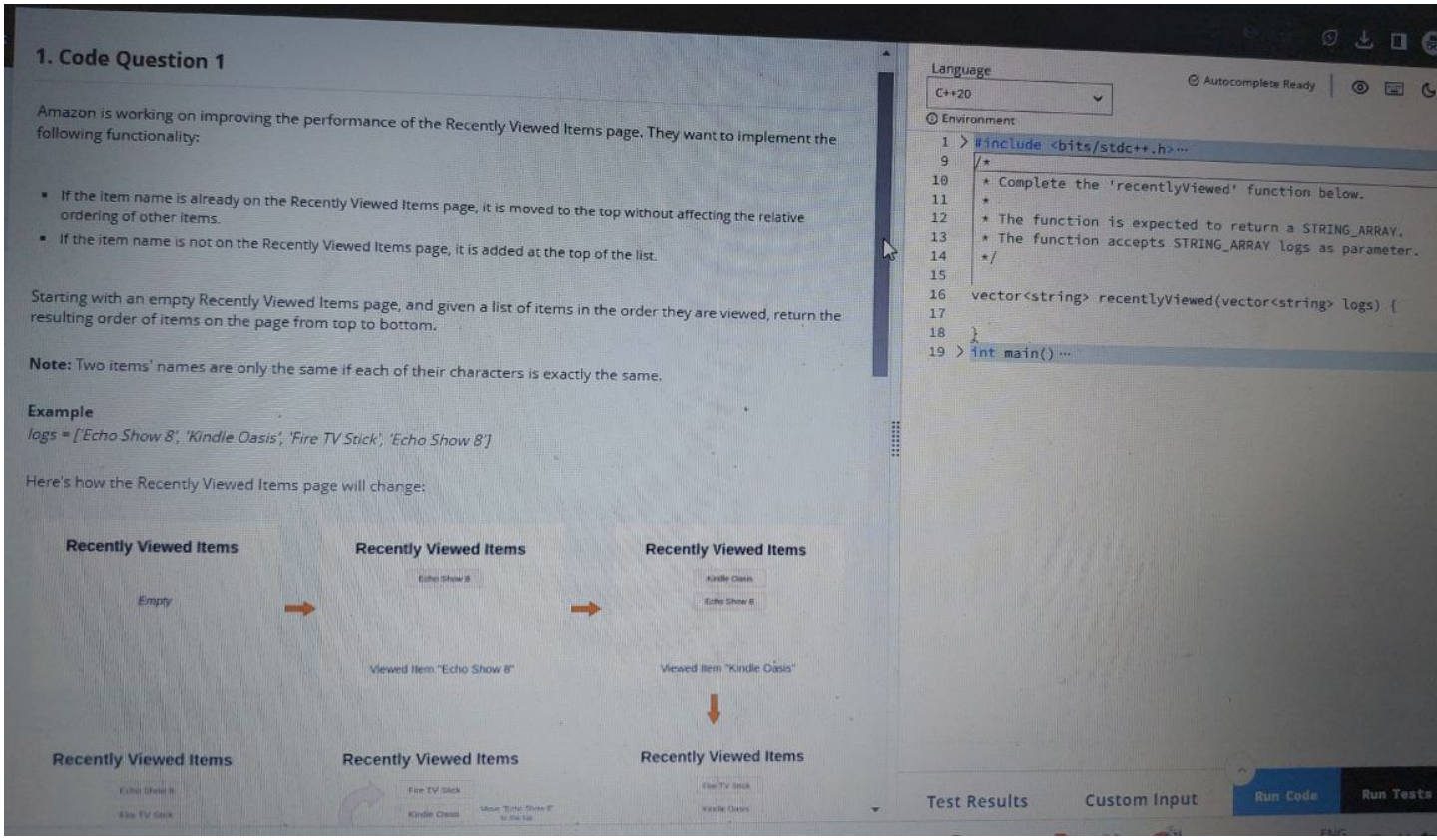

1. Code Question 1
Amazon is working on improving the performance of the Recently Viewed Items page. They want to implement the following functionality:
If the item name is already on the Recently Viewed Items page, it is moved to the top without affecting the relative ordering of other items.
If the item name is not on the Recently Viewed Items page, it is added at the top of the list.
Starting with an empty Recently Viewed Items page, and given a list of items in the order they are viewed, return the resulting order of items on the page from top to bottom.
Note: Two items' names are only the same if each of their characters is exactly the same.
Example
logs = ['Echo Show 8', 'Kindle Oasis', 'Fire TV Stick', 'Echo Show 8']
Here's how the Recently Viewed Items page will change:
Recently Viewed Items
Empty
Recently Viewed Items
Viewed item "Echo Show 8"
Recently Viewed Items
Viewed item "Kindle Oasis"
Recently Viewed Items
Recently Viewed Items
Recently Viewed Items
Language
C++20
Autocomplete Ready
Environment
#include <bits/stdc++.h>
/*
* Complete the 'recentlyViewed' function below.
*
* The function is expected to return a STRING_ARRAY.
* The function accepts STRING_ARRAY logs as parameter.
*/
vector<string> recentlyViewed(vector<string> logs) {
}
int main()
Test Results
Custom Input
Run Code
Run Tests

2. Code Question 2

Alexa is Amazon's virtual AI assistant. It makes it easy to set up your Alexa-enabled devices, listen to music, get weather updates, and much more. The team is working on a new feature that suggests days for camping based on the weather forecast.

According to a survey, a day is *ideal* for camping if the amount of rainfall has been *non-increasing* for the prior *k* days from the considered day, and will be *non-decreasing* for the following *k* days from the considered day. Given the predicted rainfall for the next *n* days, find all the ideal days. Formally, the day *i* is ideal if the following is true:

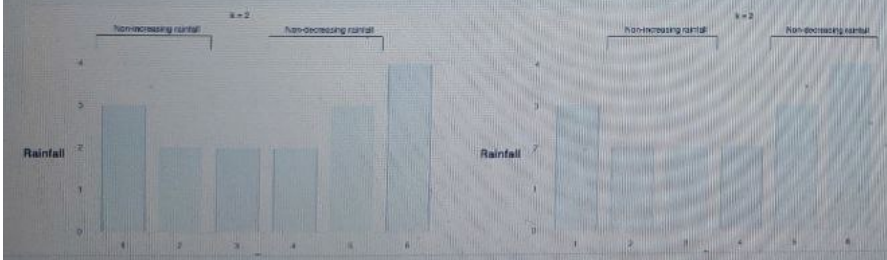
$day[i-k] \geq day[i-k+1] \geq .... \geq day[i-1] \geq day[i] \leq day[i+1] \leq .... \leq day[i+k-1] \leq day[i+k]$

Return the array of ideal days in ascending order. Note that the i^{th} element of the array represents the data for the day i + 1. It is guaranteed that the there is at least one ideal day.

Example

day = [3, 2, 2, 2, 3, 4]
k = 2

For a day to be ideal, the amount of rainfall has to be non-increasing for the prior 2 days and non-decreasing for the following 2 days.



Language
C++20
Environment
Autocomplete Ready

```
1  #include <bits/stdc++.h>...
9
10 /*
11  * Complete the 'predictDays' function below.
12  *
13  * The function is expected to return an INTEGER_A
14  * The function accepts following parameters:
15  *  1. INTEGER_ARRAY day
16  *  2. INTEGER k
17  */
18
19 vector<int> predictDays(vector<int> day, int k) {
20
21 }
22
23 int main()...
```

Test Results | Custom Input | Run Code | Run

81° ENG