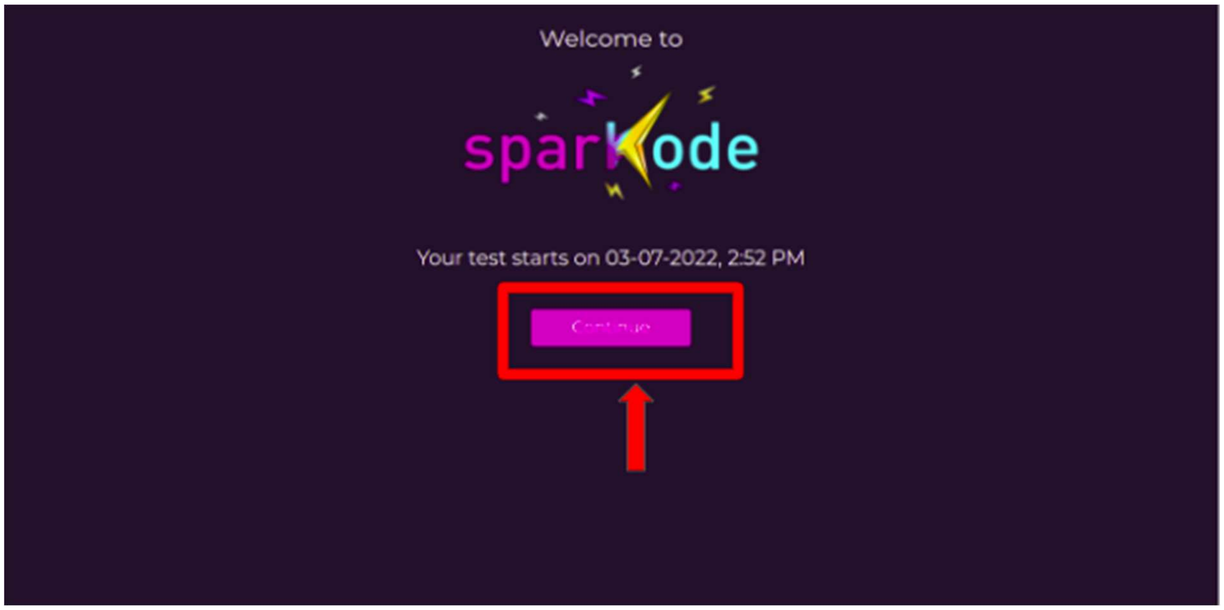
Instructions for Candidates

Step 1:

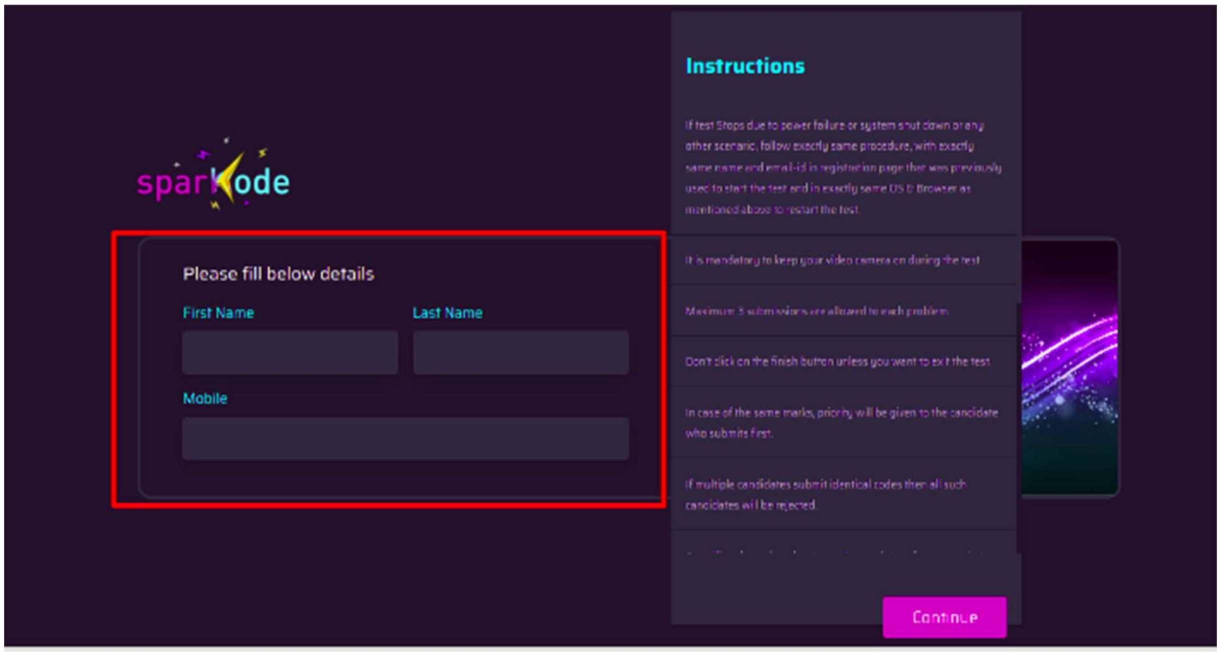
After clicking the invite link that was sent to the user via email, the screen shown below will appear with the date and time and a button to continue.



Your test starts on 03-07-2022, 2:52 PM

Step 2:

The test screen opens Click on the Continue button to go to the next screen.



Instructions

If test Stops due to power failure or system shut down or any other scenario, follow exactly same procedure, with exactly same name and email-id in registration page that was previously used to start the test and in exactly same OS & Browser as mentioned above to restart the test.

It is mandatory to keep your video camera on during the test

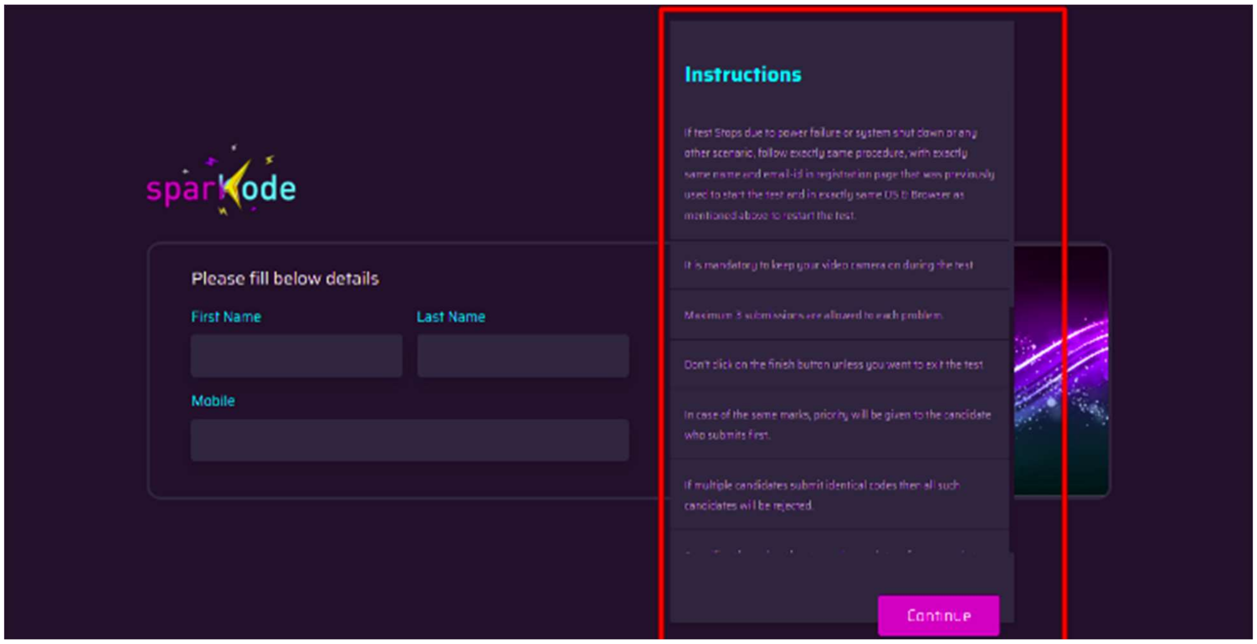
Maximum 3 submissions are allowed to each problem.

Don't click on the finish button unless you want to exit the test

In case of the same marks, priority will be given to the candidate who submits first.

If multiple candidates submit identical codes then all such candidates will be rejected.

Please fill in the details like first and last name and mobile number. The candidate should carefully read the scrollable instructions. Click Continue.



Please fill below details

First Name

Last Name

Mobile

Instructions

If test Stops due to power failure or system shut down or any other scenario, follow exactly same procedure, with exactly same name and email-id in registration page that was previously used to start the test and in exactly same OS & Browser as mentioned above to restart the test.

It is mandatory to keep your video camera on during the test

Maximum 3 submissions are allowed to each problem.

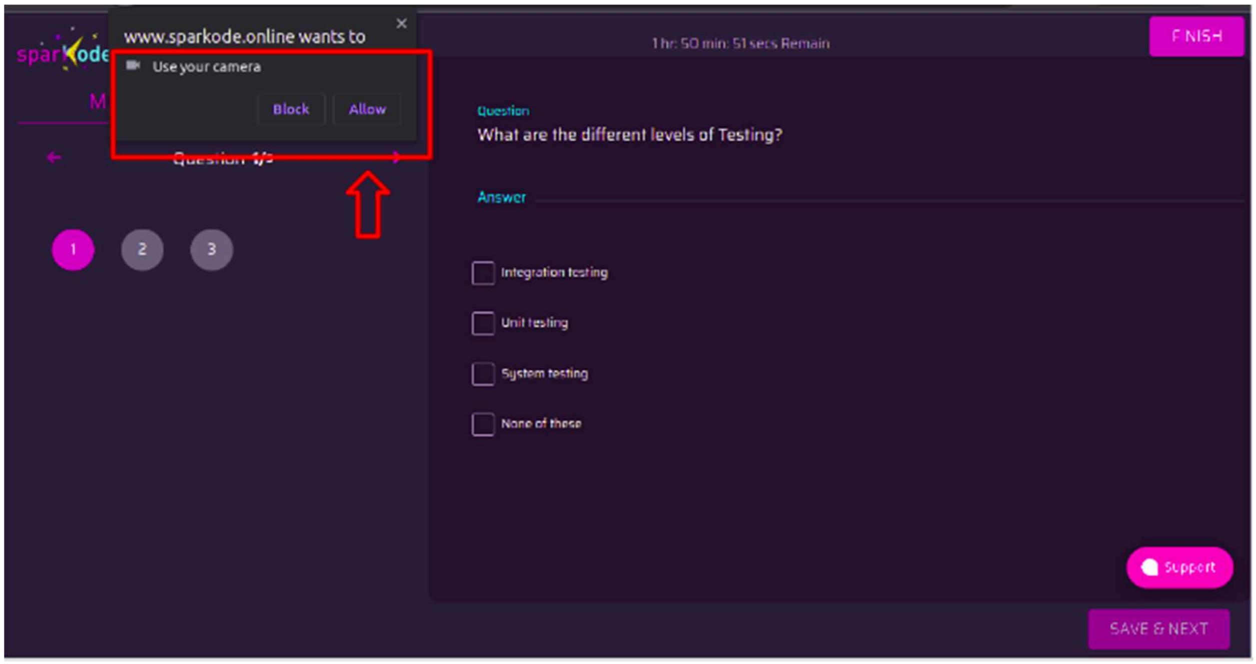
Don't click on the finish button unless you want to exit the test

In case of the same marks, priority will be given to the candidate who submits first.

If multiple candidates submit identical codes then all such candidates will be rejected.

Step 3:

To grant access to the camera—which is necessary while giving the test—click Allow.



www.sparkode.online wants to

Use your camera

1 hr: 50 min: 51 secs Remain

Question 1/3

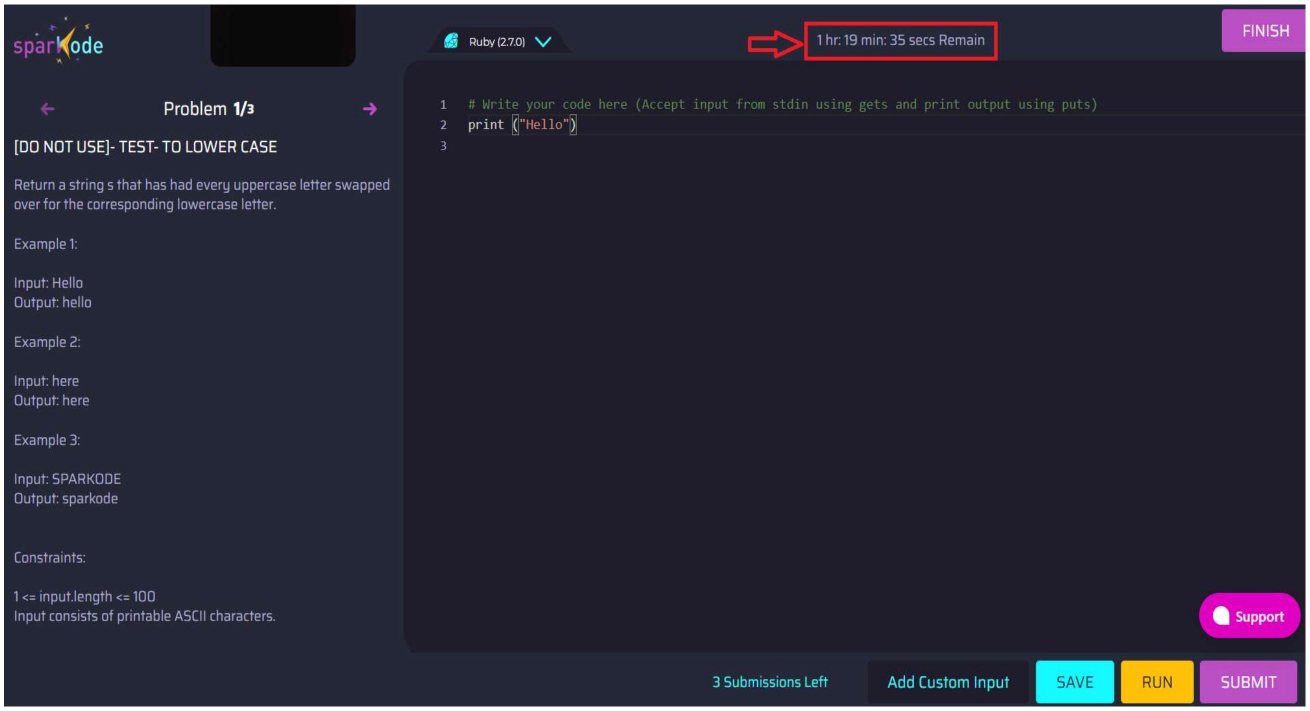
Question

What are the different levels of Testing?

Answer

- [ ] Integration testing
- [ ] Unit testing
- [ ] System testing
- [ ] None of these

The highlighted region displays the remaining test time.



[DO NOT USE]- TEST- TO LOWER CASE

Return a string s that has had every uppercase letter swapped over for the corresponding lowercase letter.

Example 1:

Input: Hello
Output: hello

Example 2:

Input: here
Output: here

Example 3:

Input: SPARKODE
Output: sparkode

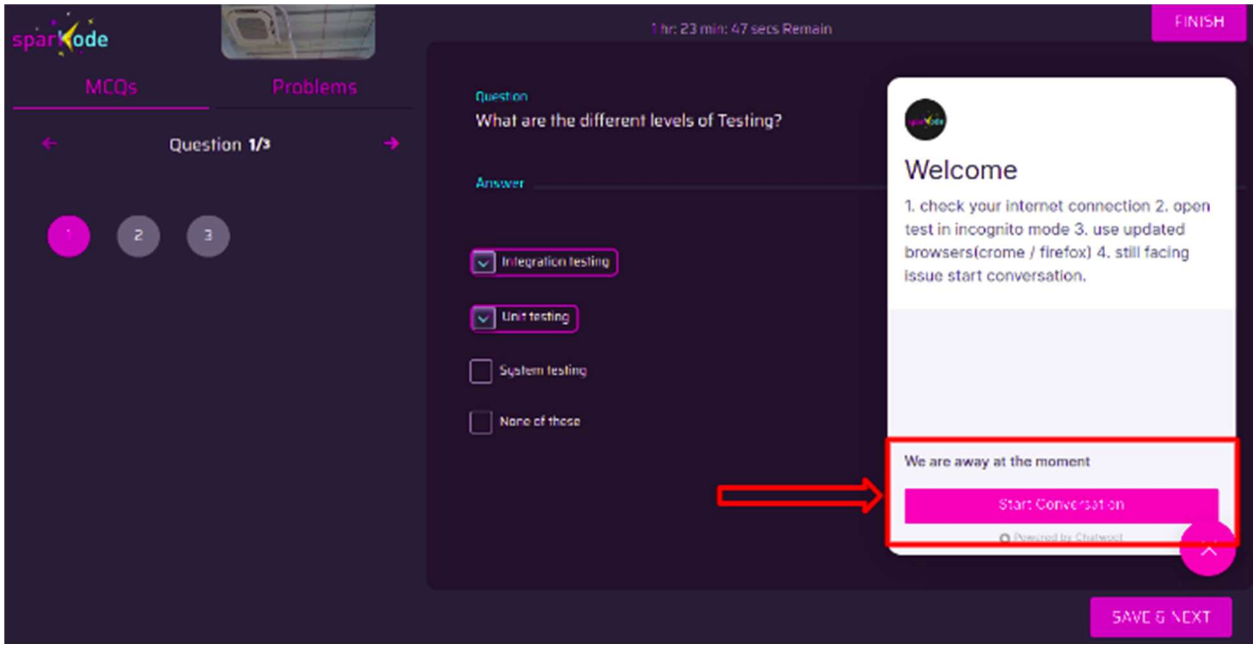
Constraints:

1 <= input.length <= 100
Input consists of printable ASCII characters.

```
# Write your code here (Accept input from stdin using gets and print output using puts)
print ("Hello")
```

Step 4:

Click on the Start Conversation button Enter the Email, Message and click on the Start Conversation button.



Question
What are the different levels of Testing?

Answer

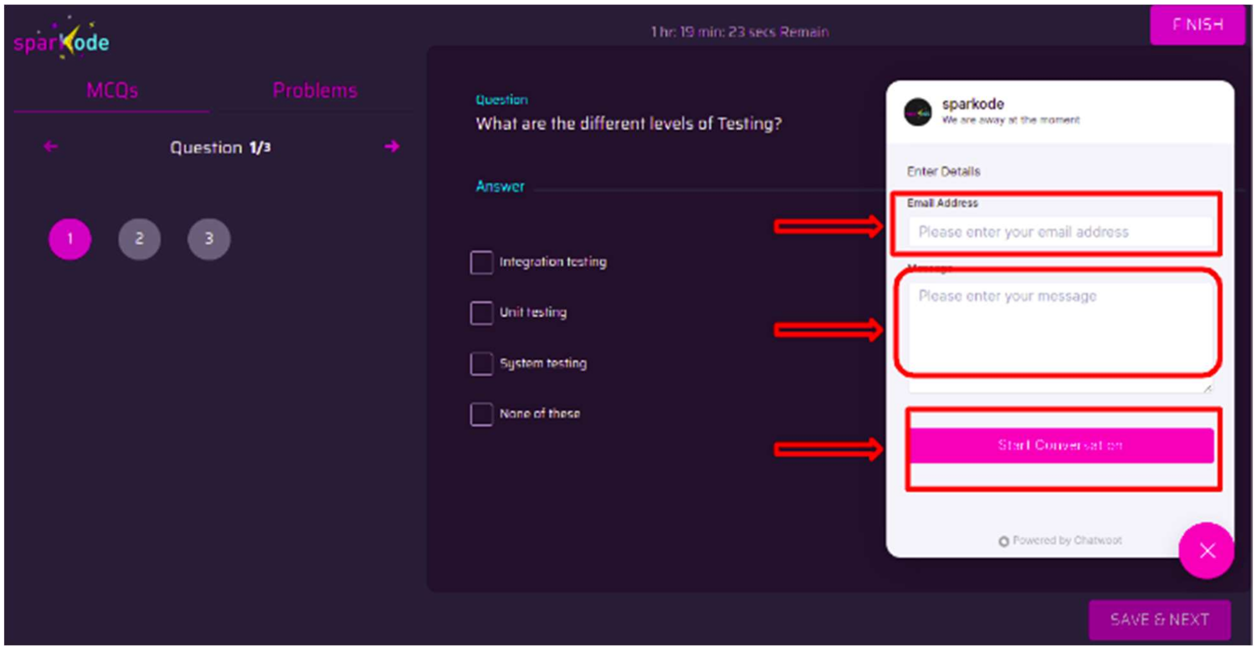
- [✓] Integration testing
- [✓] Unit testing
- [ ] System testing
- [ ] None of these

Welcome

1. check your internet connection 2. open test in incognito mode 3. use updated browsers(crome / firefox) 4. still facing issue start conversation.

We are away at the moment

Start Conversation



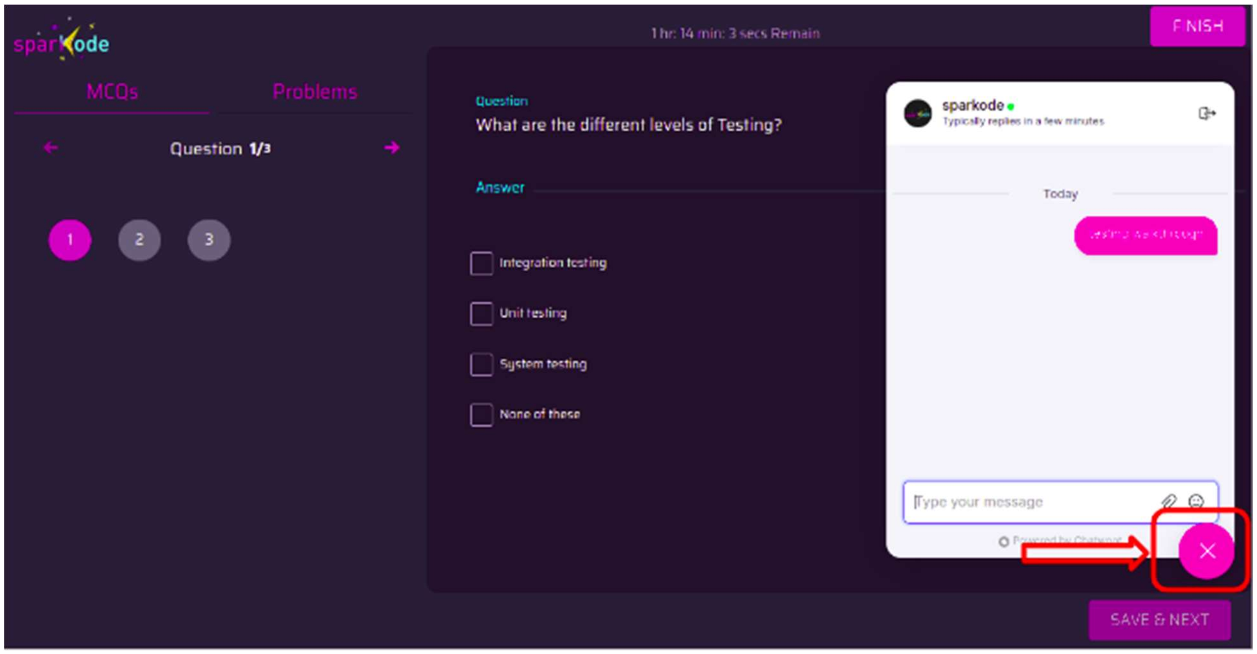
Question

What are the different levels of Testing?

Answer

- Integration testing
- Unit testing
- System testing
- None of these

Click on the ‘X’ icon to close the support chat window.



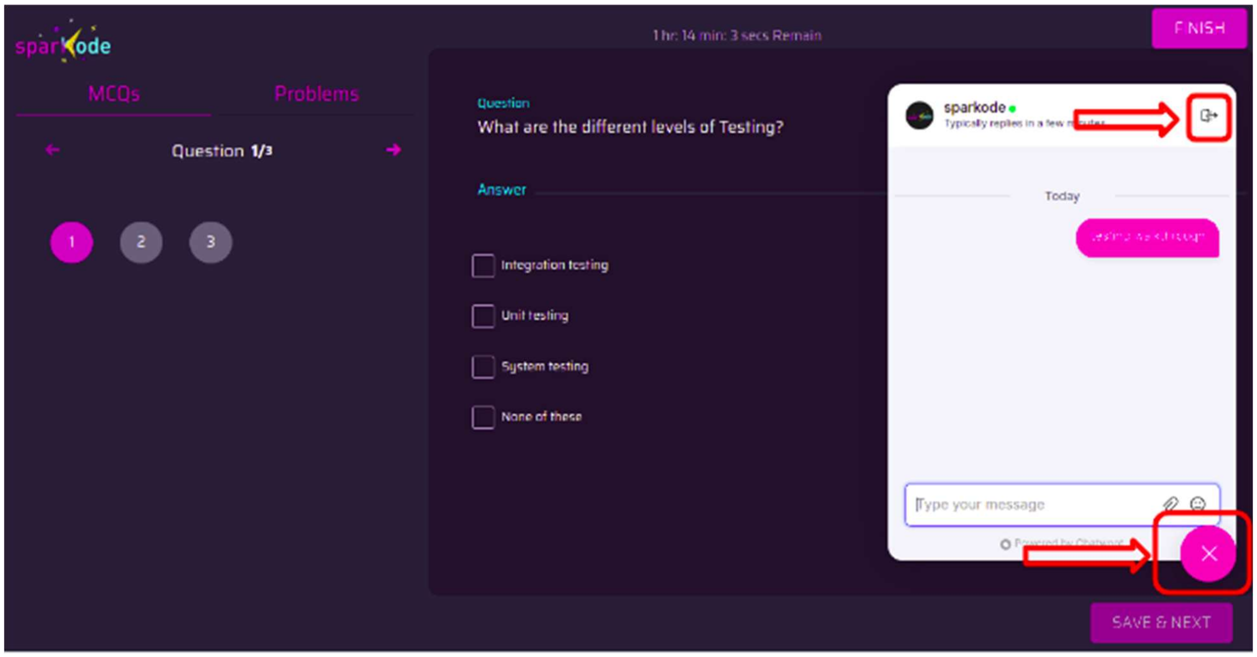
Question

What are the different levels of Testing?

Answer

- Integration testing
- Unit testing
- System testing
- None of these

Click on the top right corner of the chat window to end the conversation.



Question

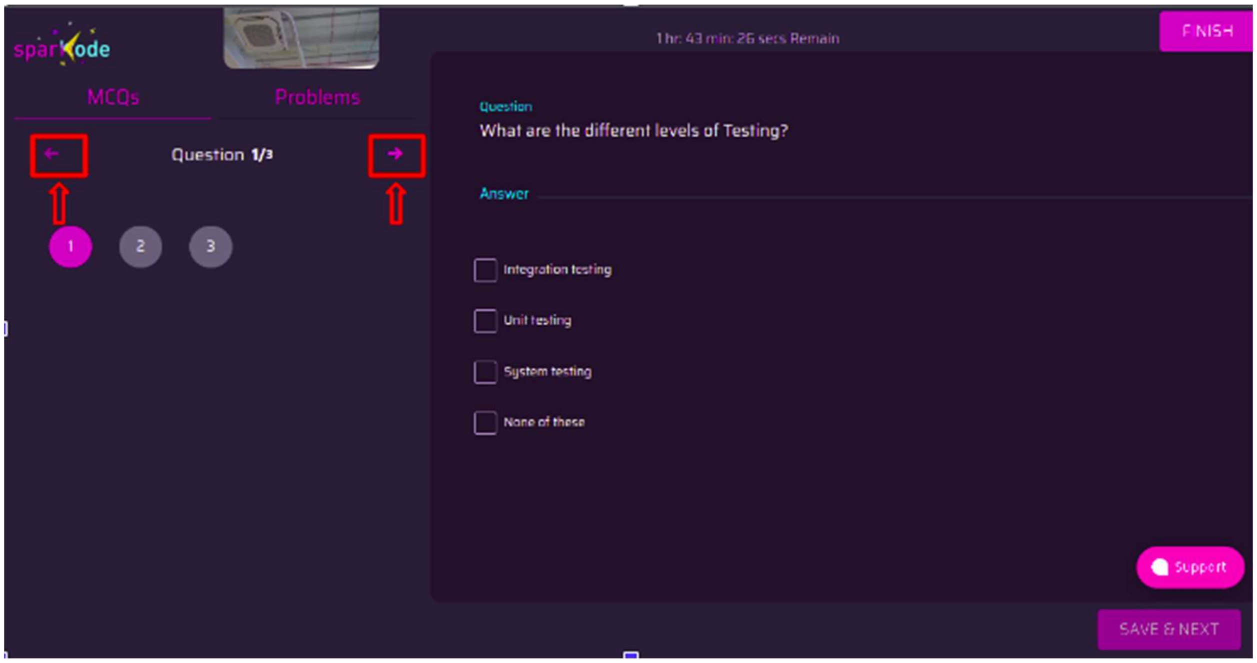
What are the different levels of Testing?

Answer

- [ ] Integration testing
- [ ] Unit testing
- [ ] System testing
- [ ] None of these

Step 5:

Click on the arrows shown to navigate between the Questions.



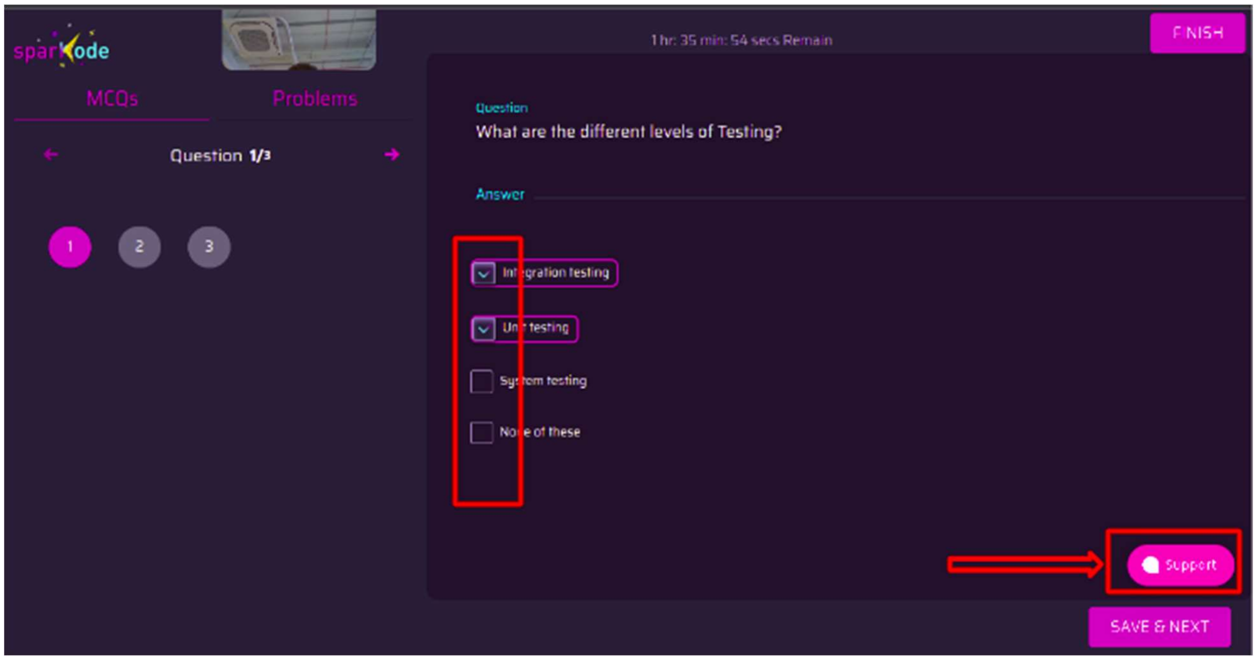
Question

What are the different levels of Testing?

Answer

- [ ] Integration testing
- [ ] Unit testing
- [ ] System testing
- [ ] None of these

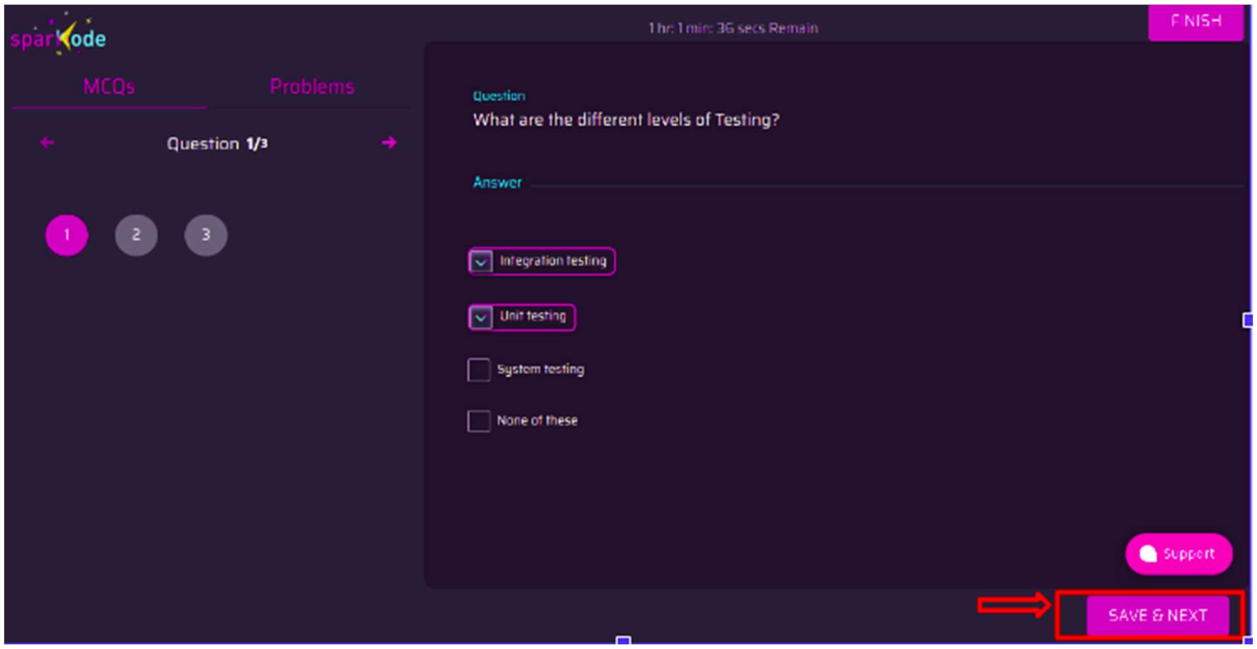
In the part where you provide your answers, check the appropriate boxes. To initiate a chat with the support staff in the event of any problems, click the Support button.



Question
What are the different levels of Testing?
Answer
[✓] Integration testing
[✓] Unit testing
[ ] System testing
[ ] None of these

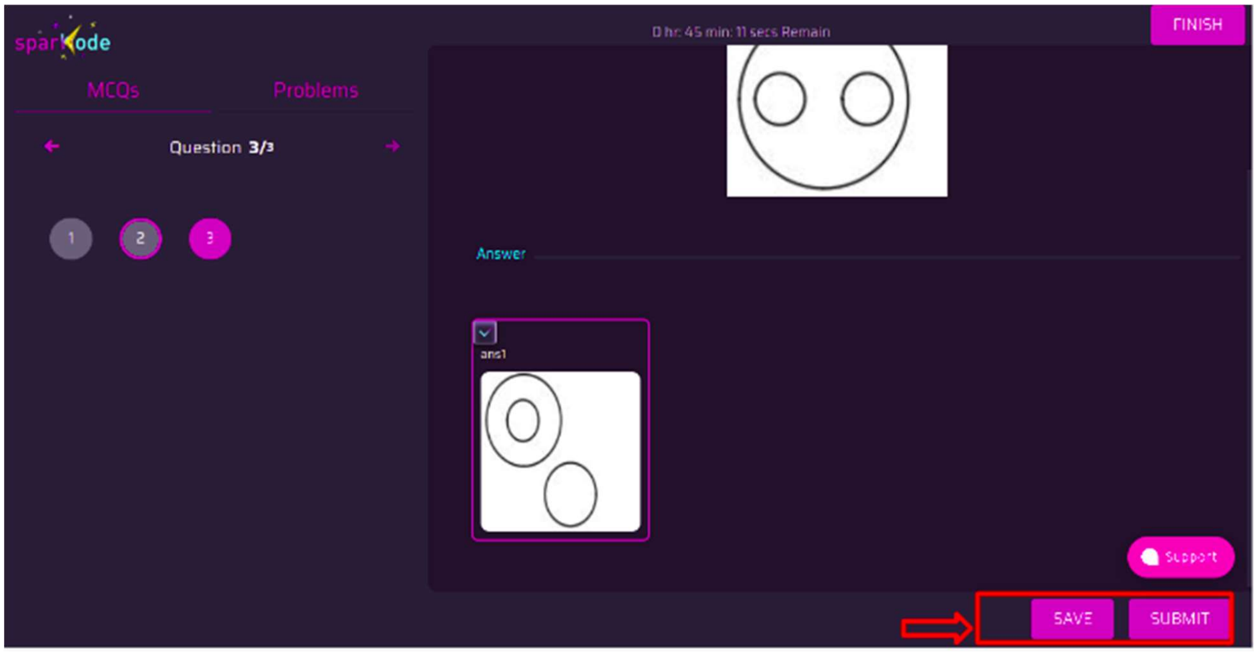
Step 6:

To save the selected response and move on to the following question, click the Save & Next option.

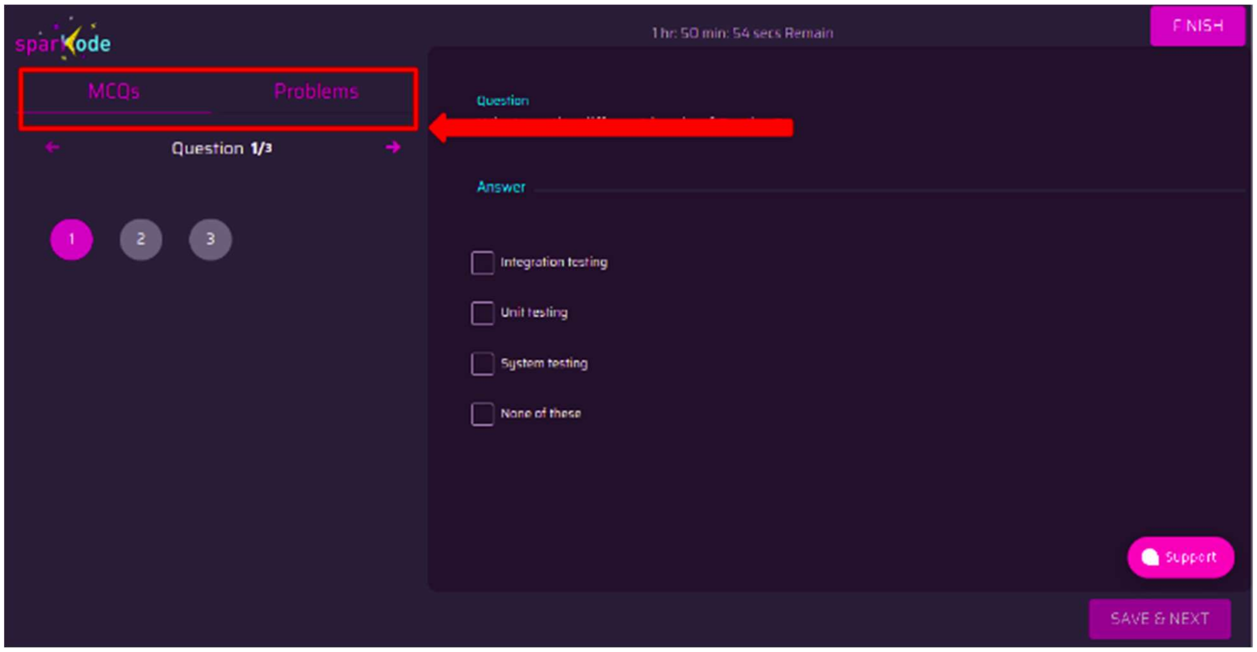


Question
What are the different levels of Testing?
Answer
[✓] Integration testing
[✓] Unit testing
[ ] System testing
[ ] None of these

Click on the Save and then Click on the Submit button to submit MCQs.

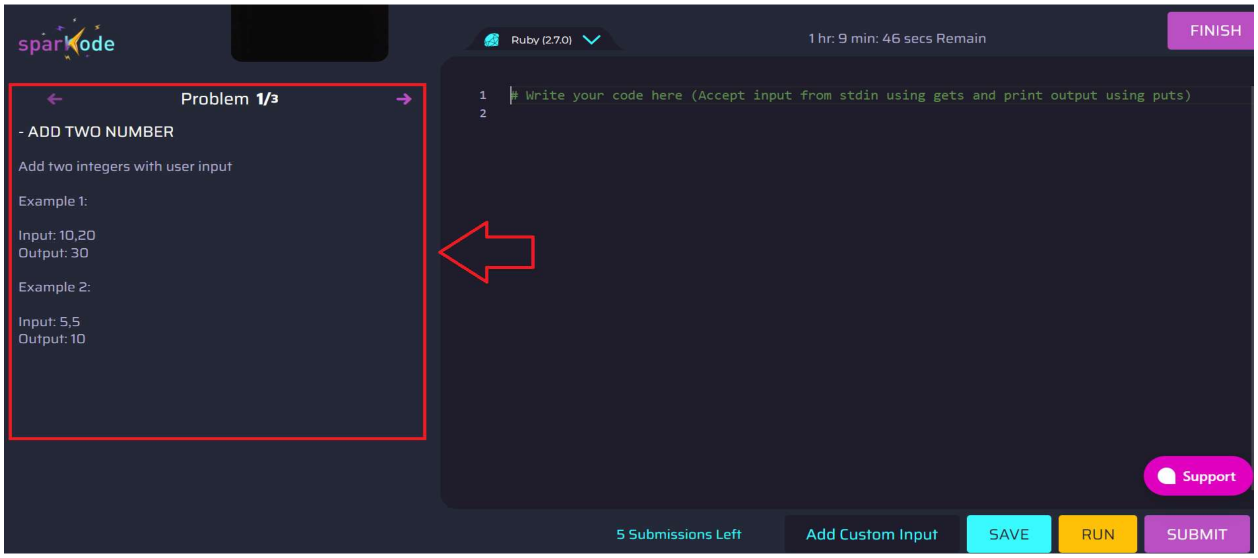


To switch between the Question categories, candidates can click on MCQs and Problems.



Step 7:

The problem statement is displayed in the portion that is highlighted.



- ADD TWO NUMBER

Add two integers with user input

Example 1:

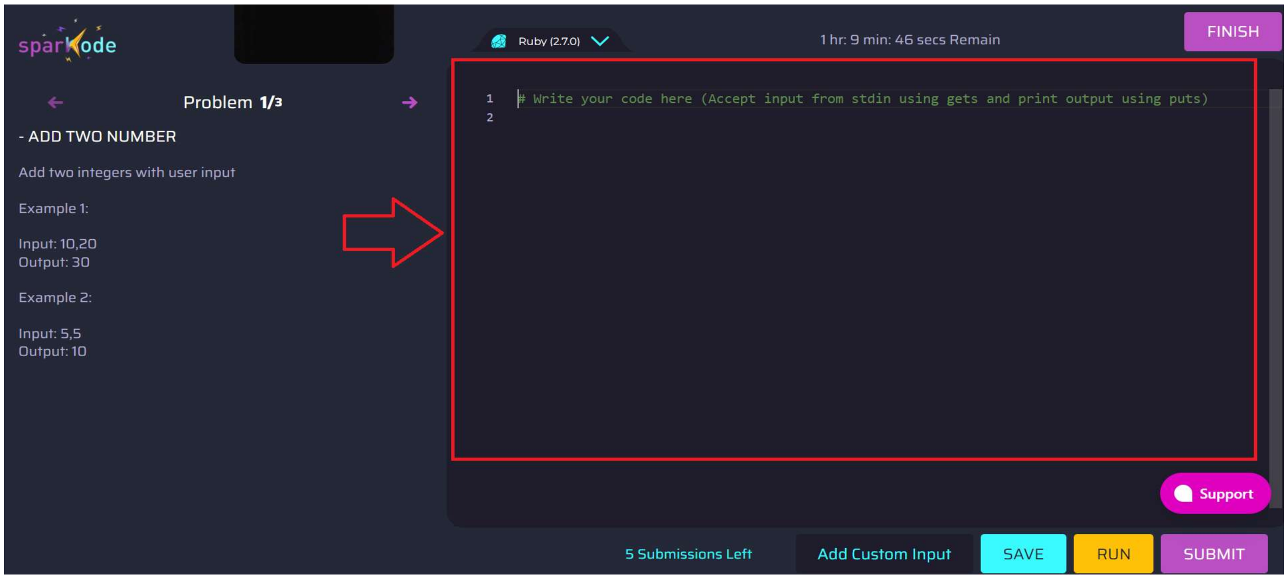
Input: 10,20
Output: 30

Example 2:

Input: 5,5
Output: 10

```
# Write your code here (Accept input from stdin using gets and print output using puts)
```

The Editor in which the candidate can write his code is displayed in the highlighted session.



- ADD TWO NUMBER

Add two integers with user input

Example 1:

Input: 10,20
Output: 30

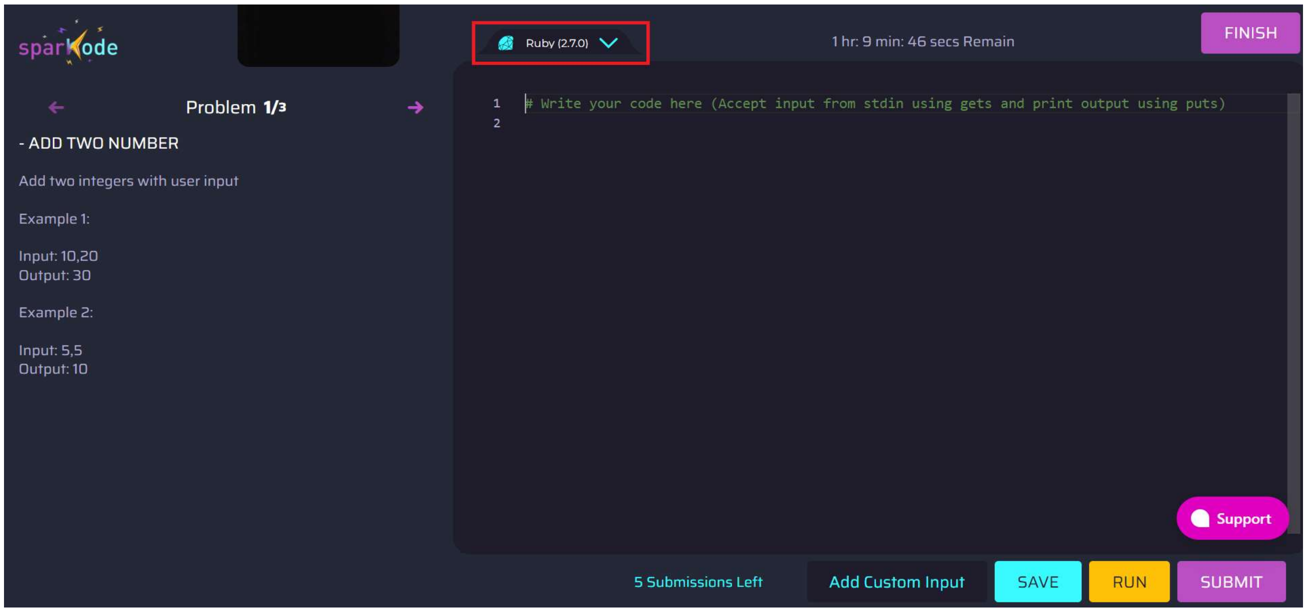
Example 2:

Input: 5,5
Output: 10

```
# Write your code here (Accept input from stdin using gets and print output using puts)
```

Step 8:

To change the writing language, click on the language template.



- ADD TWO NUMBER

Add two integers with user input

Example 1:

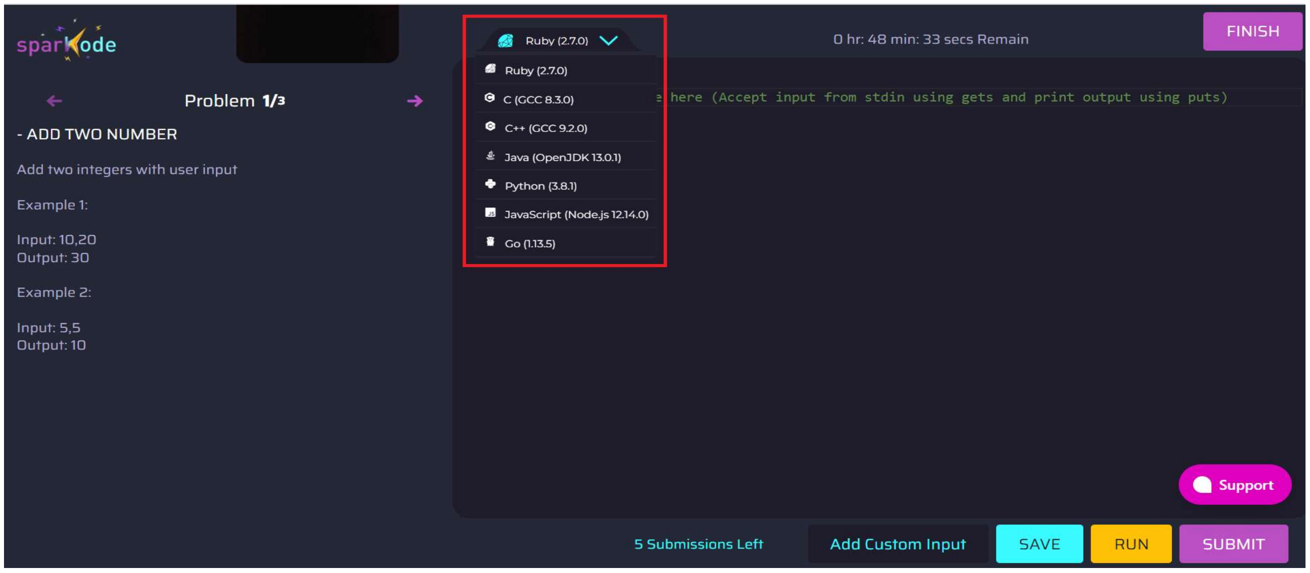
Input: 10,20
Output: 30

Example 2:

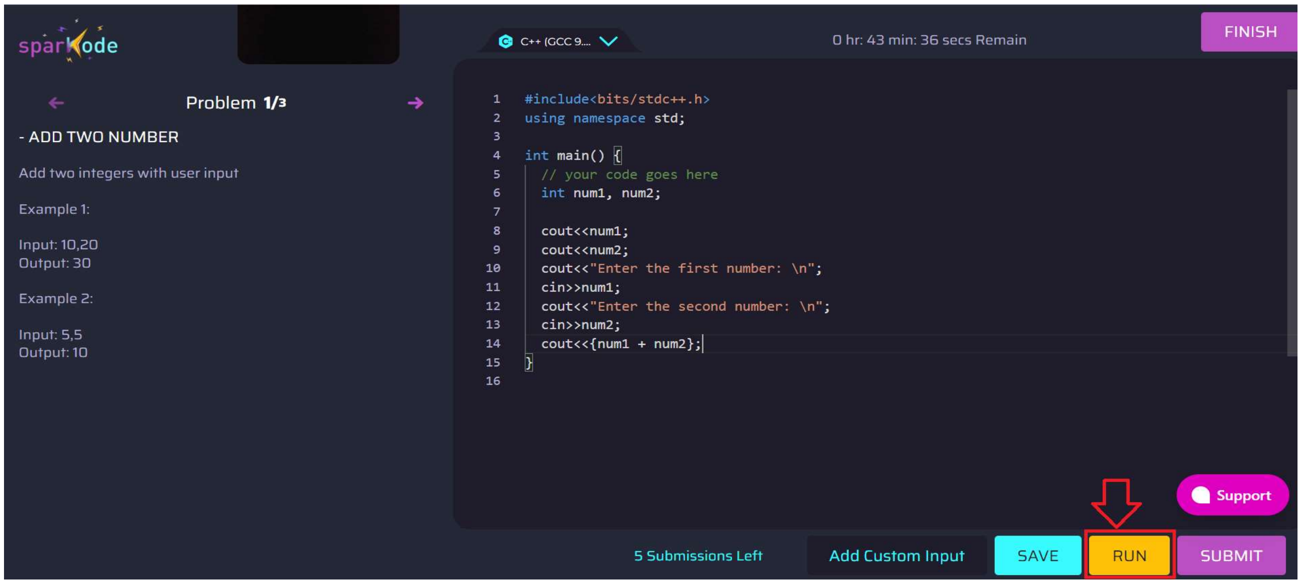
Input: 5,5
Output: 10

```
# Write your code here (Accept input from stdin using gets and print output using puts)
```

From the languages list drop-down, choose the necessary language template.



Click on the RUN button after entering the code to check the Output.



Problem 1/3

- ADD TWO NUMBER

Add two integers with user input

Example 1:

Input: 10,20
Output: 30

Example 2:

Input: 5,5
Output: 10

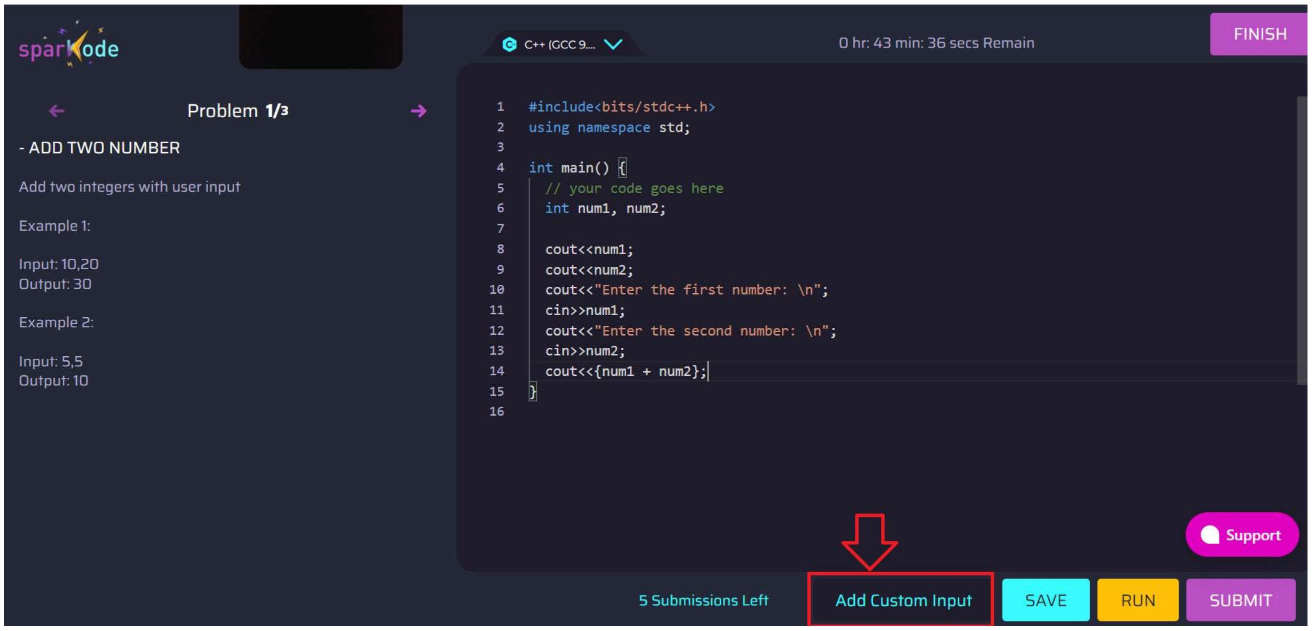
```
#include<bits/stdc++.h>
using namespace std;

int main() {
  // your code goes here
  int num1, num2;

  cout<<num1;
  cout<<num2;
  cout<<"Enter the first number: \n";
  cin>>num1;
  cout<<"Enter the second number: \n";
  cin>>num2;
  cout<<{num1 + num2};
}

```

Click on the Add Custom input to enter the custom input.



Problem 1/3

- ADD TWO NUMBER

Add two integers with user input

Example 1:

Input: 10,20
Output: 30

Example 2:

Input: 5,5
Output: 10

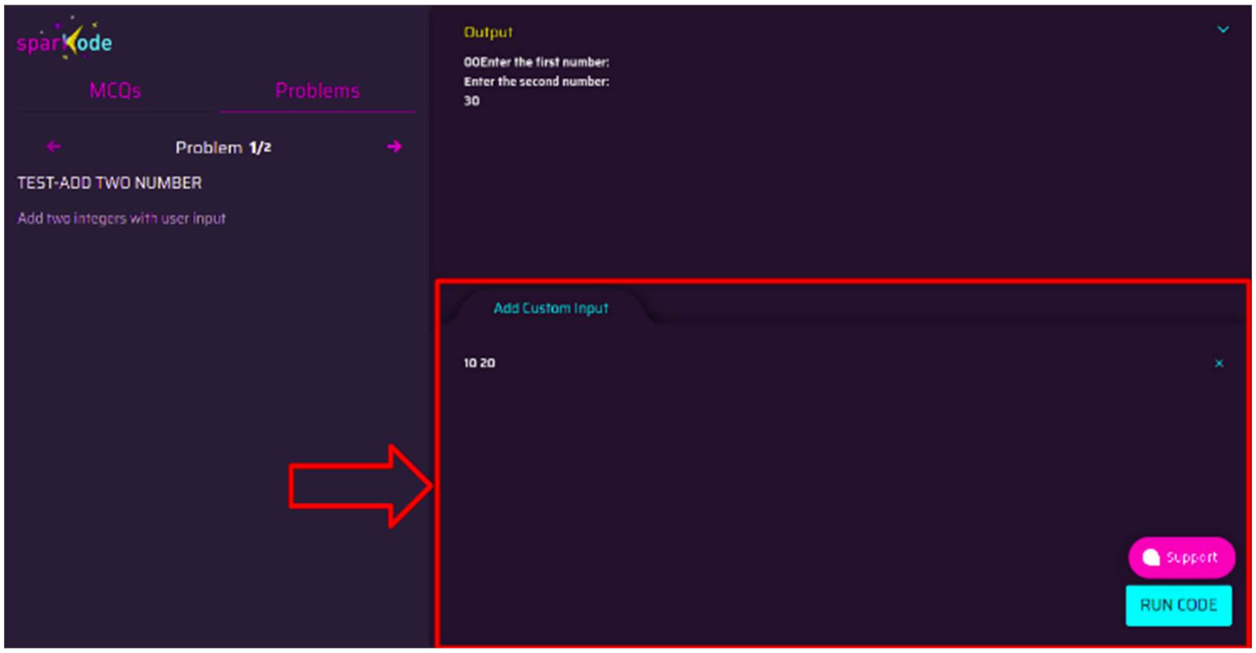
```
#include<bits/stdc++.h>
using namespace std;

int main() {
  // your code goes here
  int num1, num2;

  cout<<num1;
  cout<<num2;
  cout<<"Enter the first number: \n";
  cin>>num1;
  cout<<"Enter the second number: \n";
  cin>>num2;
  cout<<{num1 + num2};
}

```

Enter Input in the highlighted field.



TEST-ADD TWO NUMBER

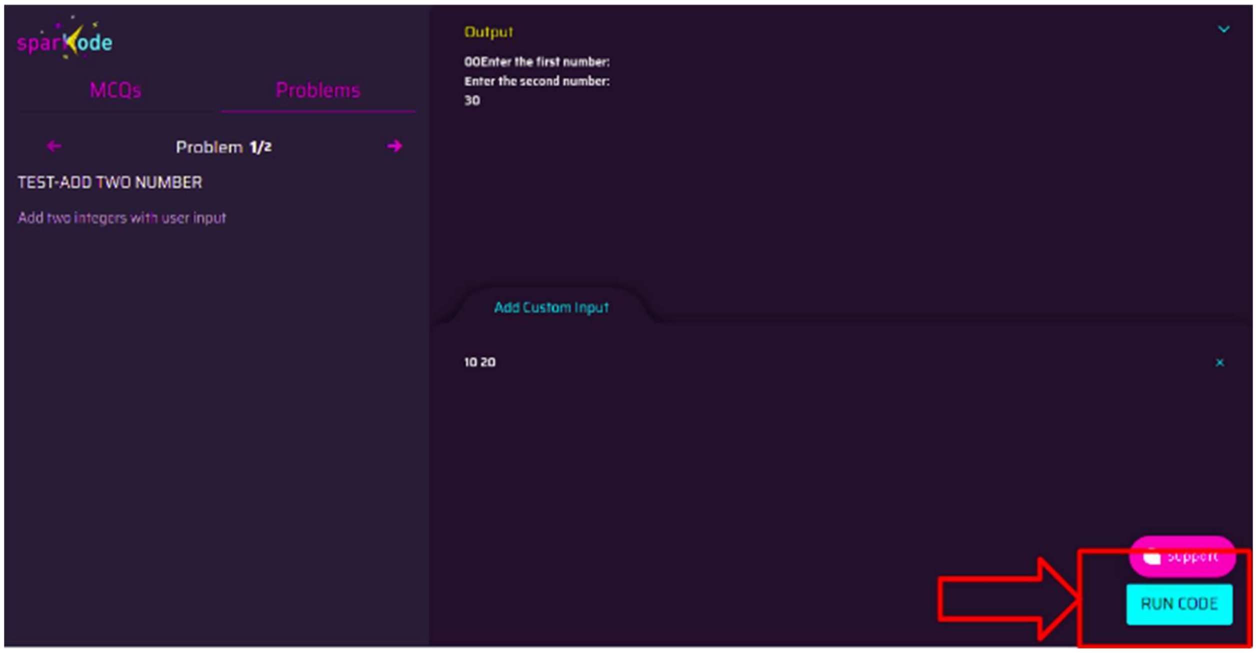
Add two integers with user input

Output

```
00Enter the first number:
Enter the second number:
30
```

```
10 20
```

Click on the Run Code to check the Output.



TEST-ADD TWO NUMBER

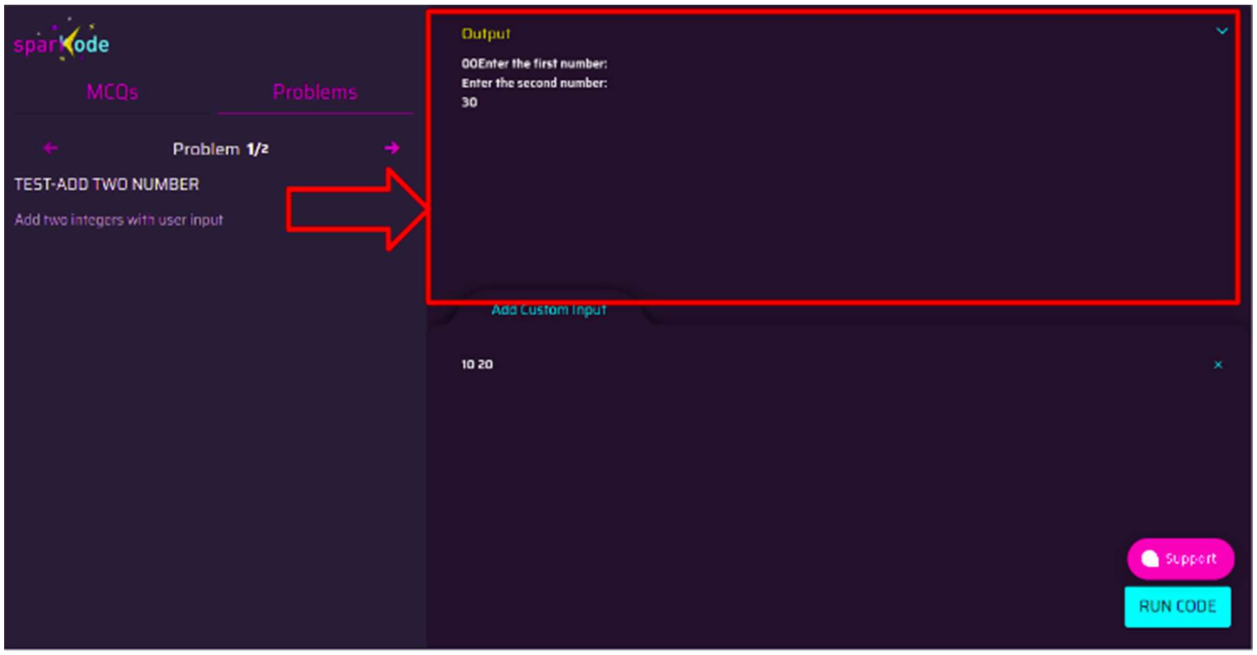
Add two integers with user input

Output

```
00Enter the first number:
Enter the second number:
30
```

```
10 20
```

Output can be shown in the highlighted section.



TEST-ADD TWO NUMBER

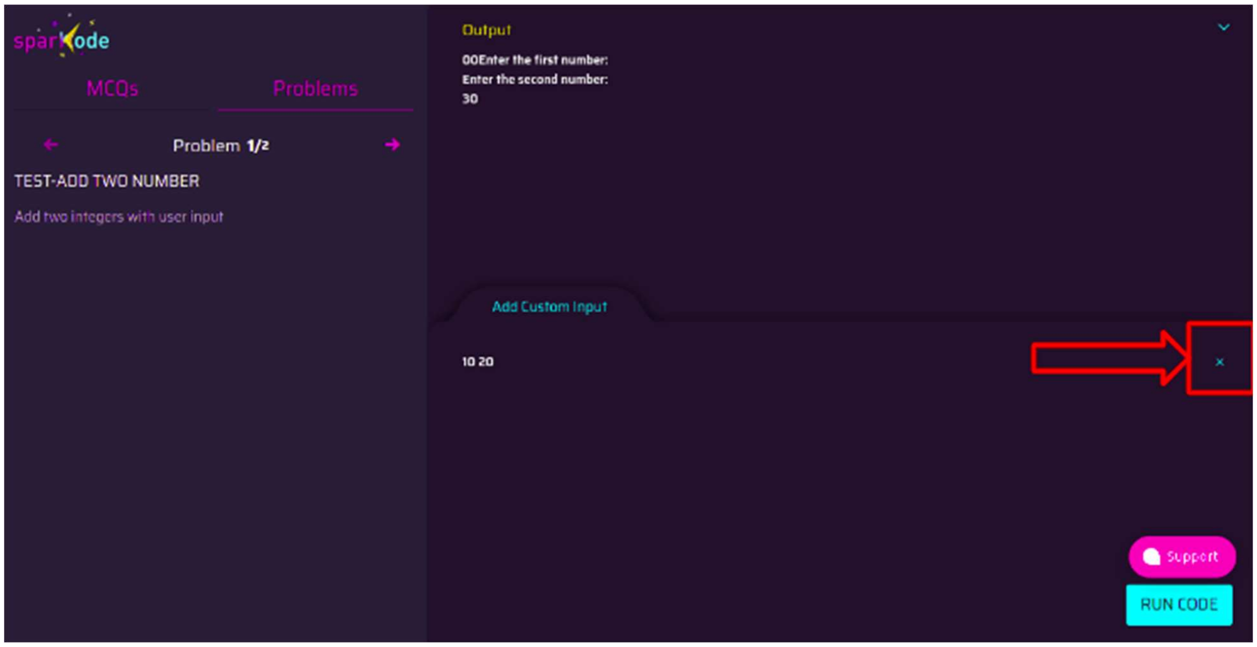
Add two integers with user input

Output

```
00Enter the first number:
Enter the second number:
30
```

```
10 20
```

Click on the 'x' icon to close the custom input window.



TEST-ADD TWO NUMBER

Add two integers with user input

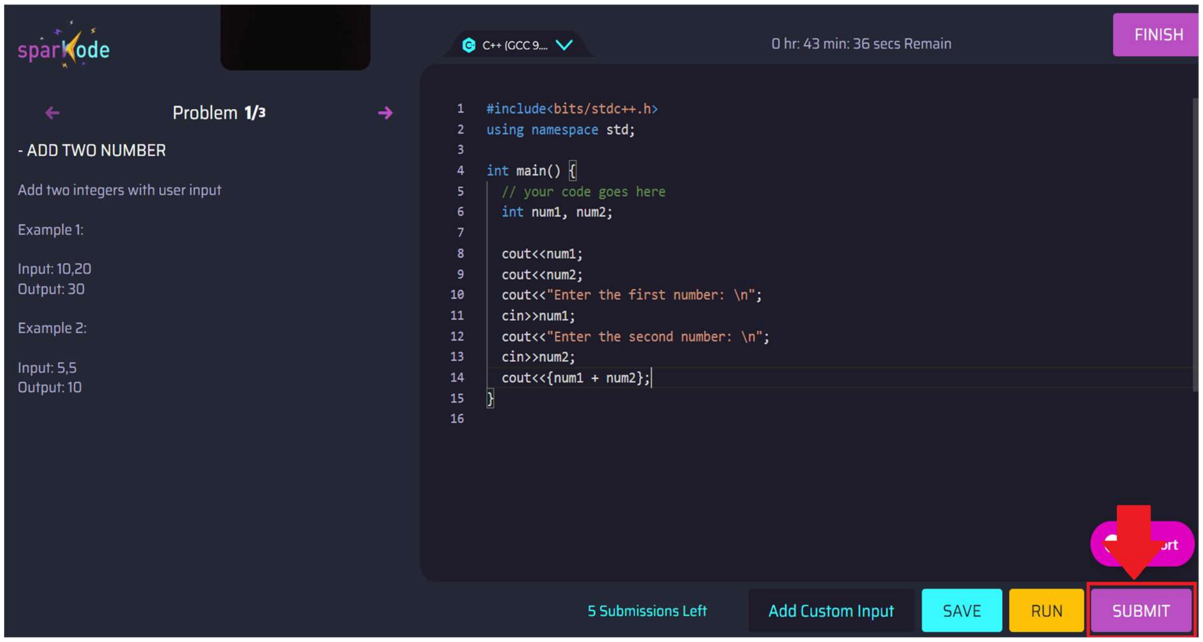
Output

```
00Enter the first number:
Enter the second number:
30
```

```
10 20
```

Step 9:

Click on the Submit button to submit the code and run the test cases.



- ADD TWO NUMBER

Add two integers with user input

Example 1:

Input: 10,20
Output: 30

Example 2:

Input: 5,5
Output: 10

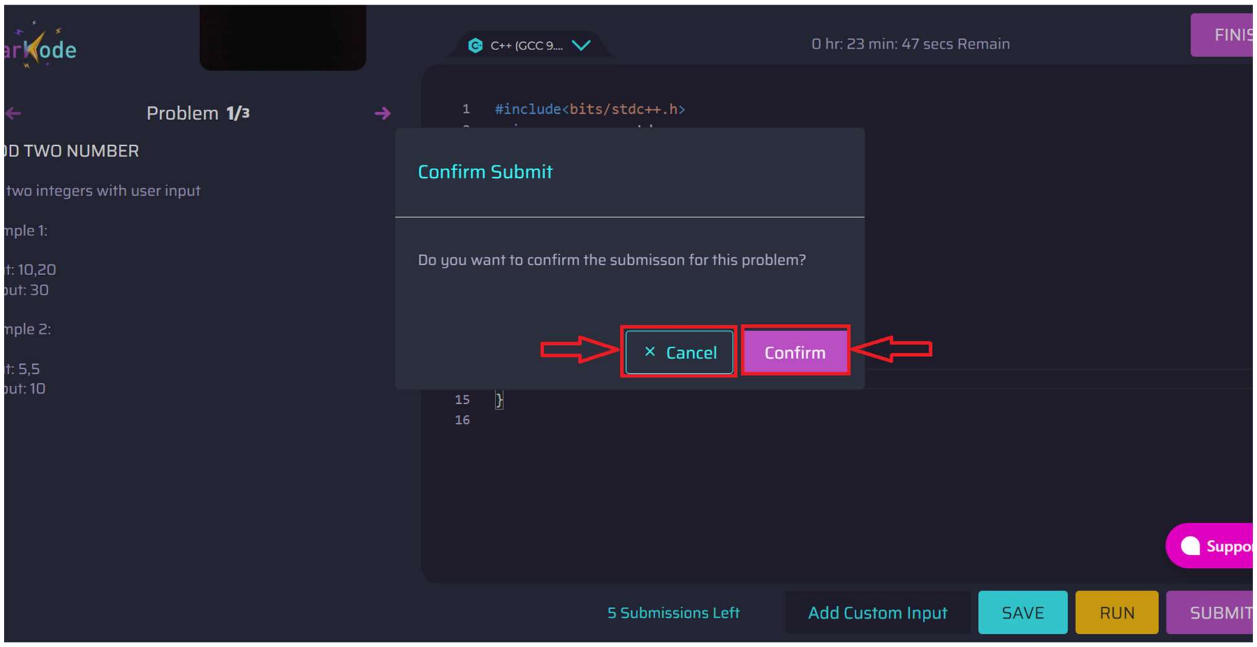
```
#include<bits/stdc++.h>
using namespace std;

int main() {
  // your code goes here
  int num1, num2;

  cout<<num1;
  cout<<num2;
  cout<<"Enter the first number: \n";
  cin>>num1;
  cout<<"Enter the second number: \n";
  cin>>num2;
  cout<<{num1 + num2};
}

```

Click on the Confirm button for submission of the code or click on the Cancel button to cancel the submission.

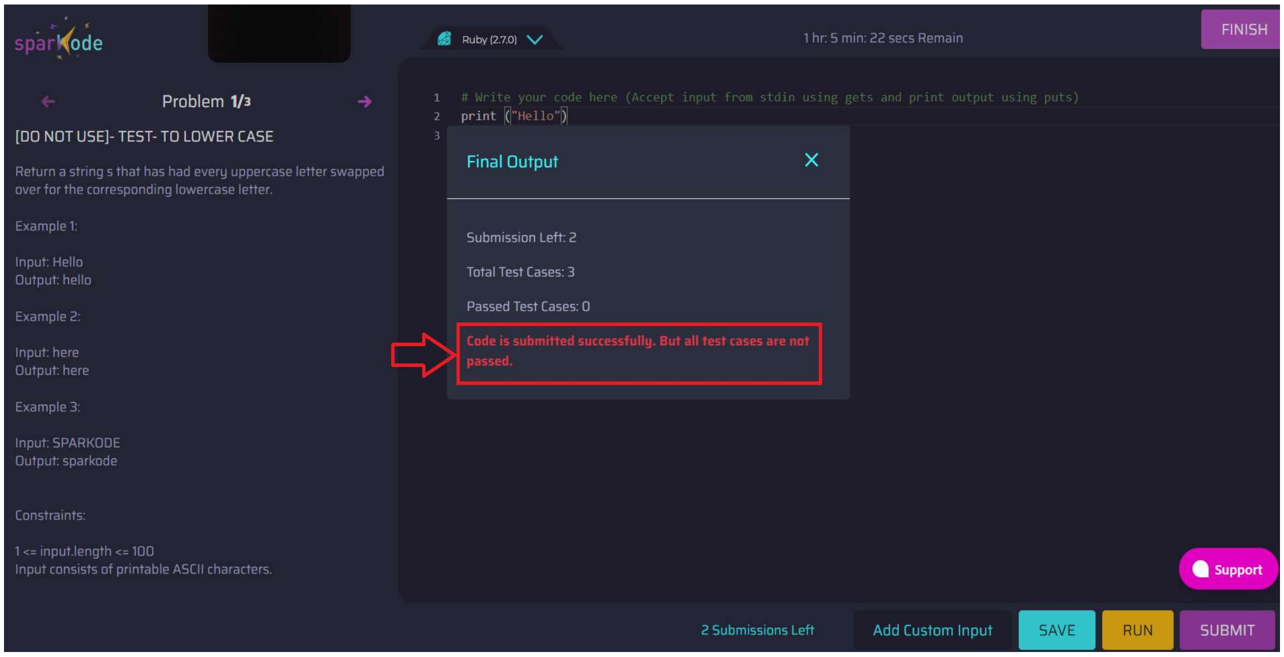


Confirm Submit

Do you want to confirm the submisson for this problem?

Step 10:

After clicking the confirm button a popup will appear showing the submission count, Total test cases number, Passed Test Cases with count and Status as Accepted or Failed. When all the test cases fail then the status will be shown as below.



[DO NOT USE]- TEST- TO LOWER CASE

Return a string s that has had every uppercase letter swapped over for the corresponding lowercase letter.

Example 1:

Input: Hello
Output: hello

Example 2:

Input: here
Output: here

Example 3:

Input: SPARKODE
Output: sparkode

Constraints:

1 <= input.length <= 100
Input consists of printable ASCII characters.

```
# Write your code here (Accept input from stdin using gets and print output using puts)
print ("Hello")
```

Final Output

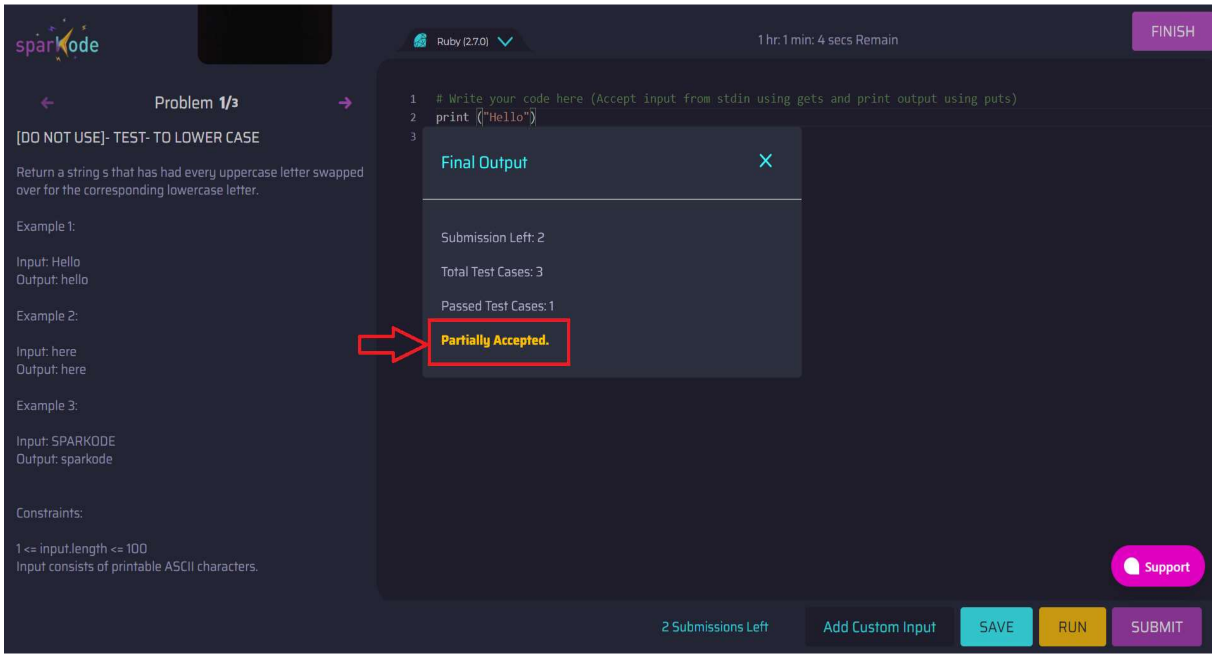
Submission Left: 2

Total Test Cases: 3

Passed Test Cases: 0

Code is submitted successfully. But all test cases are not passed.

When some of the test cases pass then the status will be shown as below.



[DO NOT USE]- TEST- TO LOWER CASE

Return a string s that has had every uppercase letter swapped over for the corresponding lowercase letter.

Example 1:

Input: Hello
Output: hello

Example 2:

Input: here
Output: here

Example 3:

Input: SPARKODE
Output: sparkode

Constraints:

1 <= input.length <= 100
Input consists of printable ASCII characters.

```
# Write your code here (Accept input from stdin using gets and print output using puts)
print ("Hello")
```

Final Output

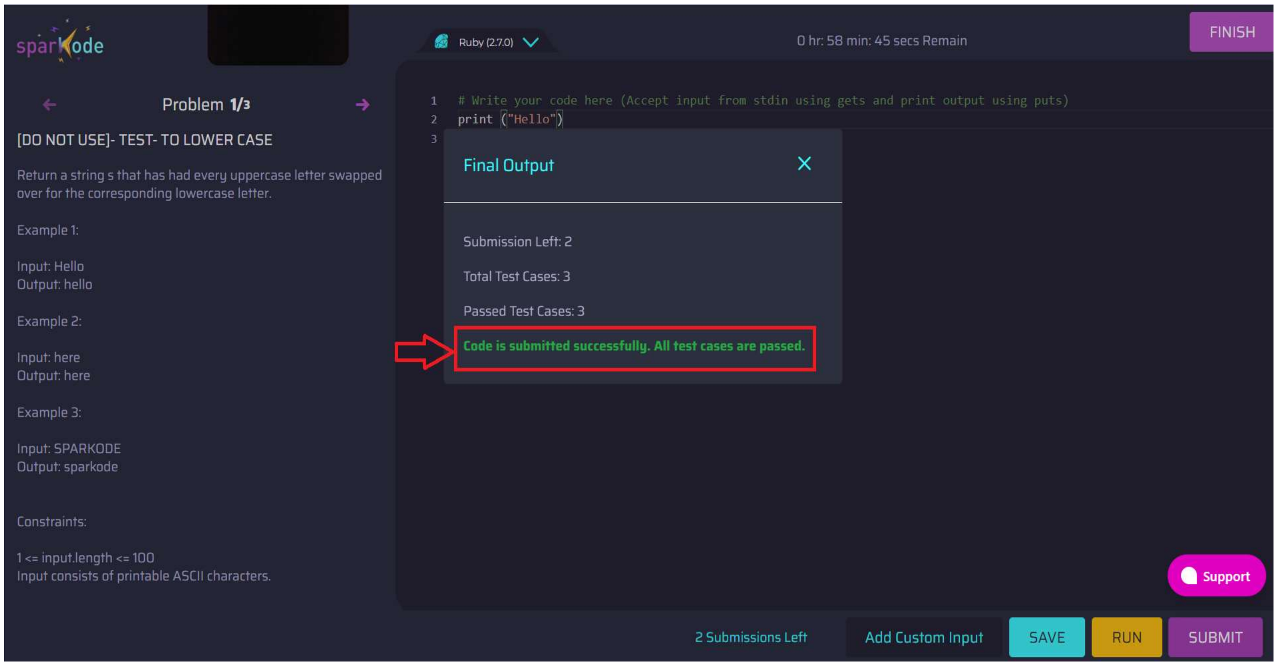
Submission Left: 2

Total Test Cases: 3

Passed Test Cases: 1

Partially Accepted.

When all the test cases pass then the status will be shown as below.



[DO NOT USE]- TEST- TO LOWER CASE

Return a string s that has had every uppercase letter swapped over for the corresponding lowercase letter.

Example 1:

Input: Hello
Output: hello

Example 2:

Input: here
Output: here

Example 3:

Input: SPARKODE
Output: sparkode

Constraints:

1 <= input.length <= 100
Input consists of printable ASCII characters.

```
# Write your code here (Accept input from stdin using gets and print output using puts)
print ("Hello")
```

Final Output

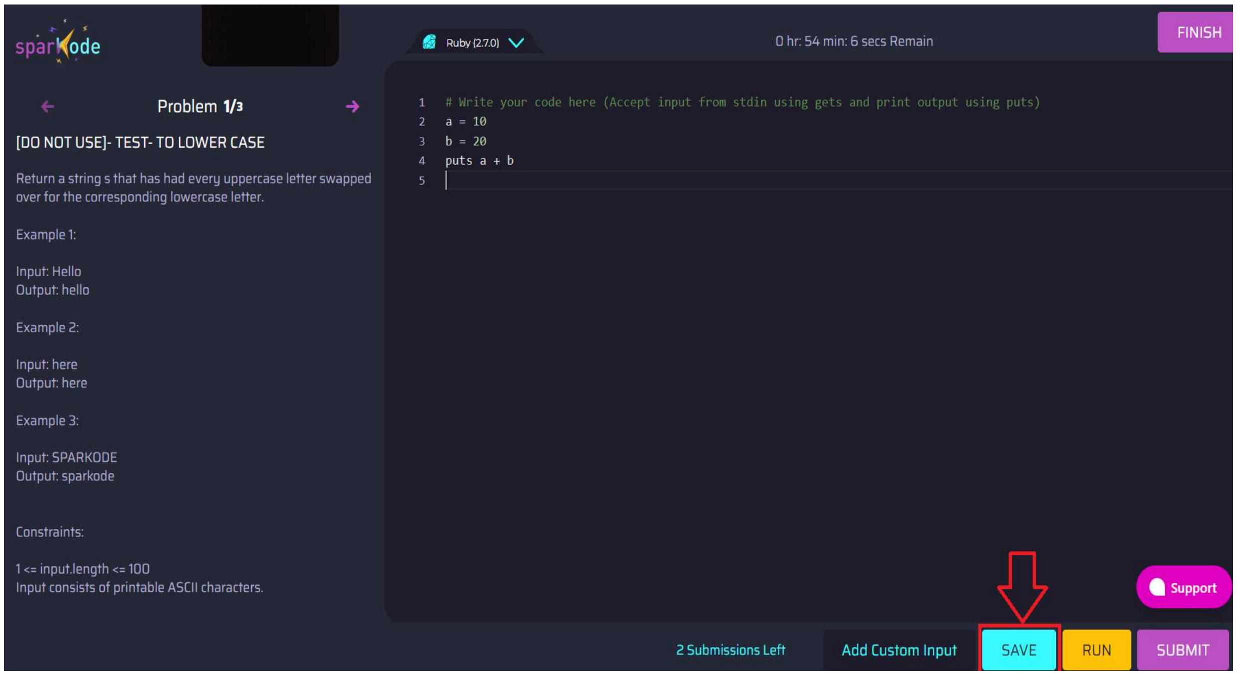
Submission Left: 2

Total Test Cases: 3

Passed Test Cases: 3

Code is submitted successfully. All test cases are passed.

Click on the Save Button to Save the written Code.



[DO NOT USE]- TEST- TO LOWER CASE

Return a string s that has had every uppercase letter swapped over for the corresponding lowercase letter.

Example 1:

Input: Hello
Output: hello

Example 2:

Input: here
Output: here

Example 3:

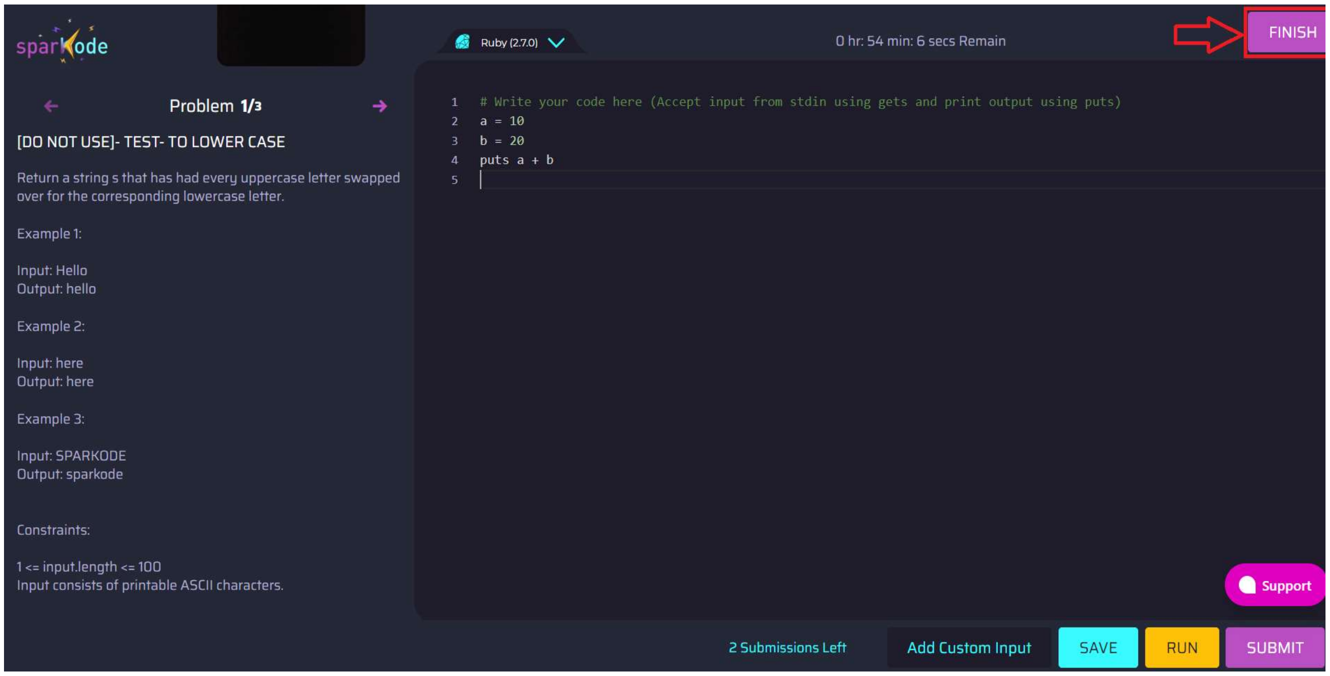
Input: SPARKODE
Output: sparkode

Constraints:

1 <= input.length <= 100
Input consists of printable ASCII characters.

```
# Write your code here (Accept input from stdin using gets and print output using puts)
a = 10
b = 20
puts a + b
```

To complete the test, click the Finish button.



[DO NOT USE]- TEST- TO LOWER CASE

Return a string s that has had every uppercase letter swapped over for the corresponding lowercase letter.

Example 1:

Input: Hello
Output: hello

Example 2:

Input: here
Output: here

Example 3:

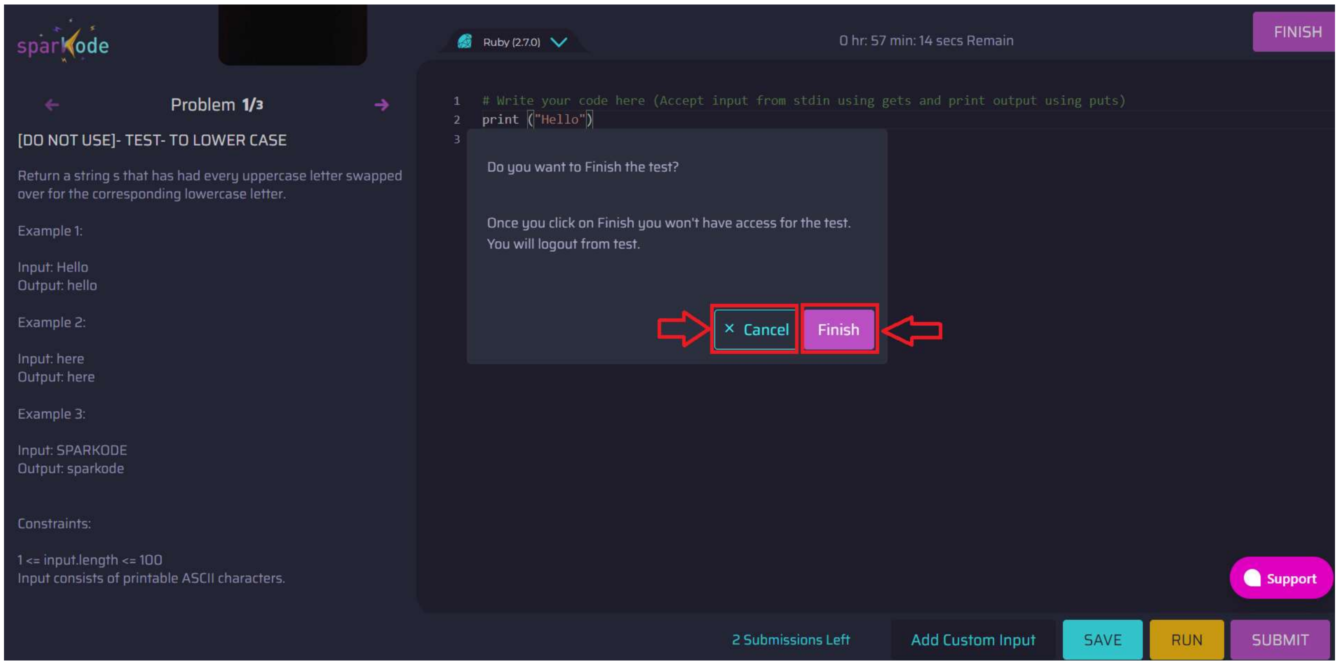
Input: SPARKODE
Output: sparkode

Constraints:

1 <= input.length <= 100
Input consists of printable ASCII characters.

```
# Write your code here (Accept input from stdin using gets and print output using puts)
a = 10
b = 20
puts a + b
```

Click on the Finish button to complete the test or click on the Cancel button to cancel the Finish action.



[DO NOT USE]- TEST- TO LOWER CASE

Return a string s that has had every uppercase letter swapped over for the corresponding lowercase letter.

Example 1:

Input: Hello
Output: hello

Example 2:

Input: here
Output: here

Example 3:

Input: SPARKODE
Output: sparkode

Constraints:

1 <= input.length <= 100
Input consists of printable ASCII characters.

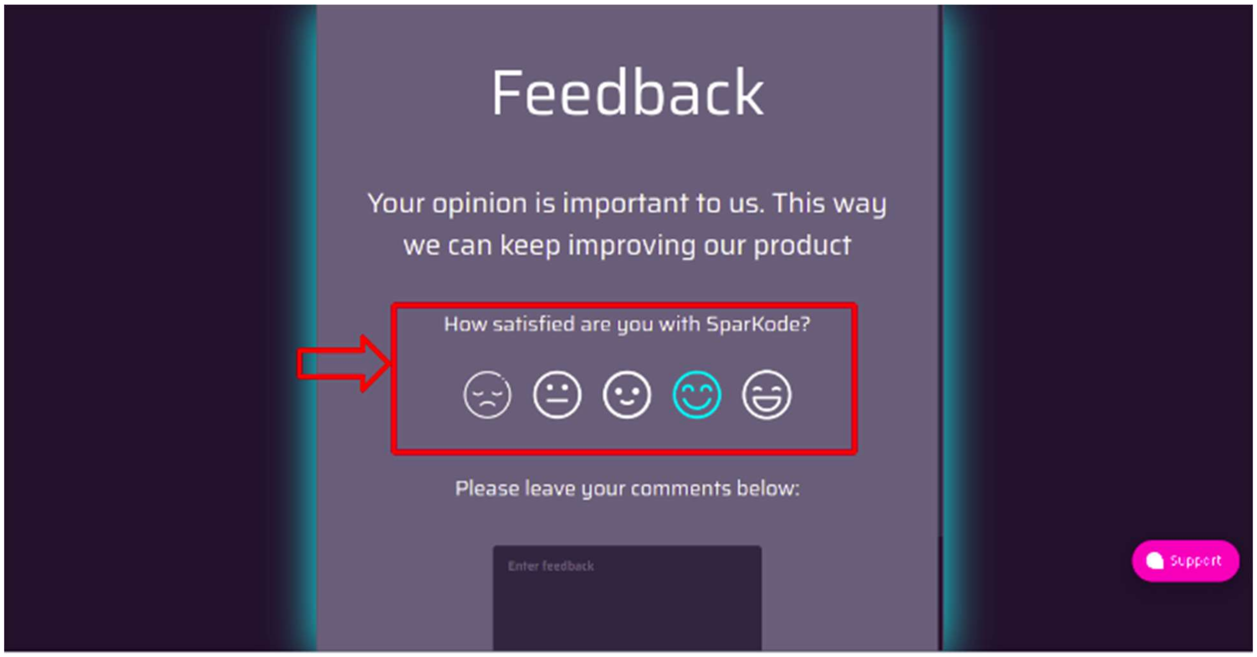
```
# Write your code here (Accept input from stdin using gets and print output using puts)
print ("Hello")
```

Do you want to Finish the test?

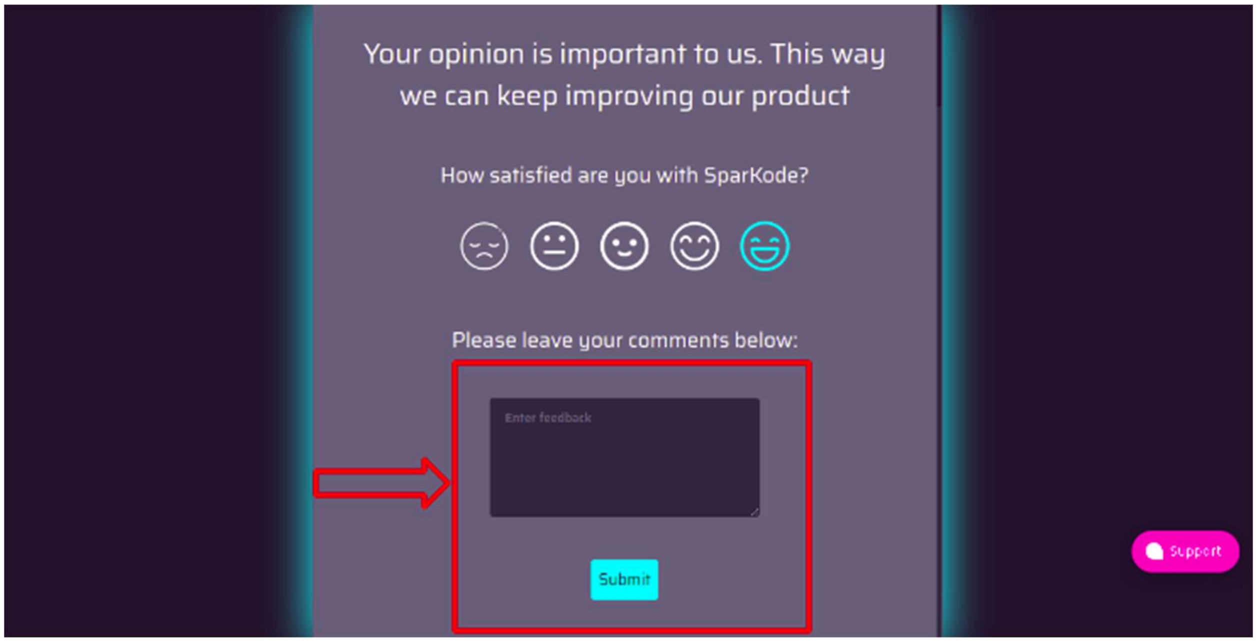
Once you click on Finish you won't have access for the test.
You will logout from test.

Step 11:

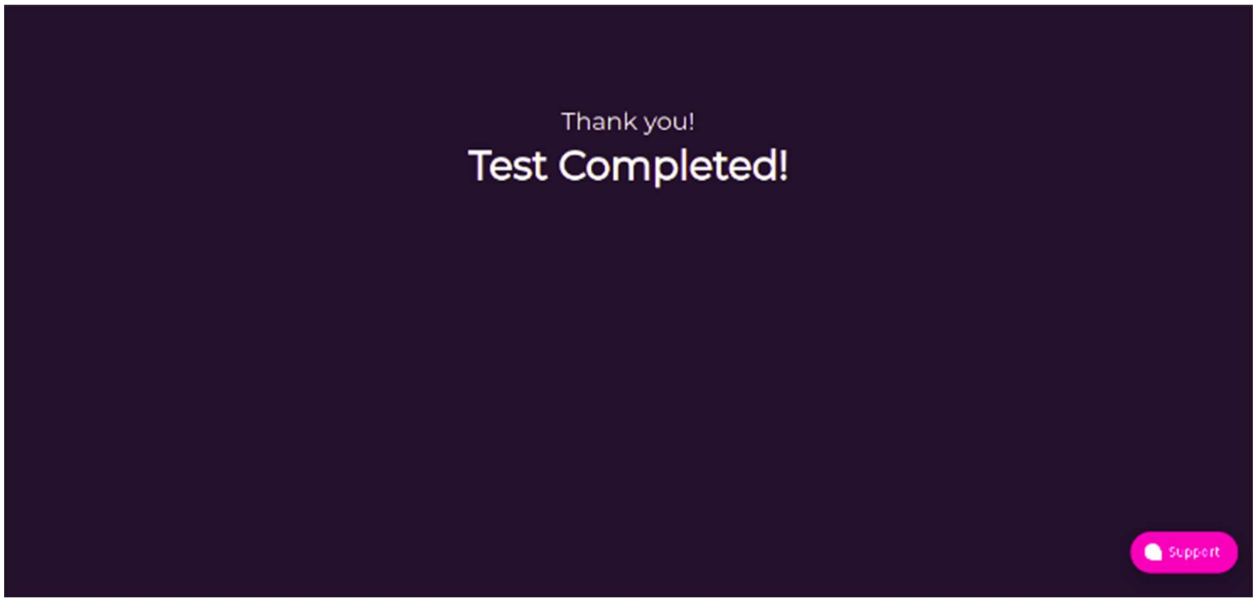
After Clicking on the Finish button candidates will be redirected to the Feedback Page where the Candidate should select the Feedback emoticon rating from the 1-5 score range.


Feedback
Your opinion is important to us. This way we can keep improving our product
How satisfied are you with SparKode?
Please leave your comments below:
Enter feedback
Support

Enter the Feedback in the text field and Click the Submit button to complete the test.


Your opinion is important to us. This way we can keep improving our product
How satisfied are you with SparKode?
Please leave your comments below:
Enter feedback
Submit
Support

Once the feedback is submitted then the Test completion page will be displayed.


Thank you!

Test Completed!