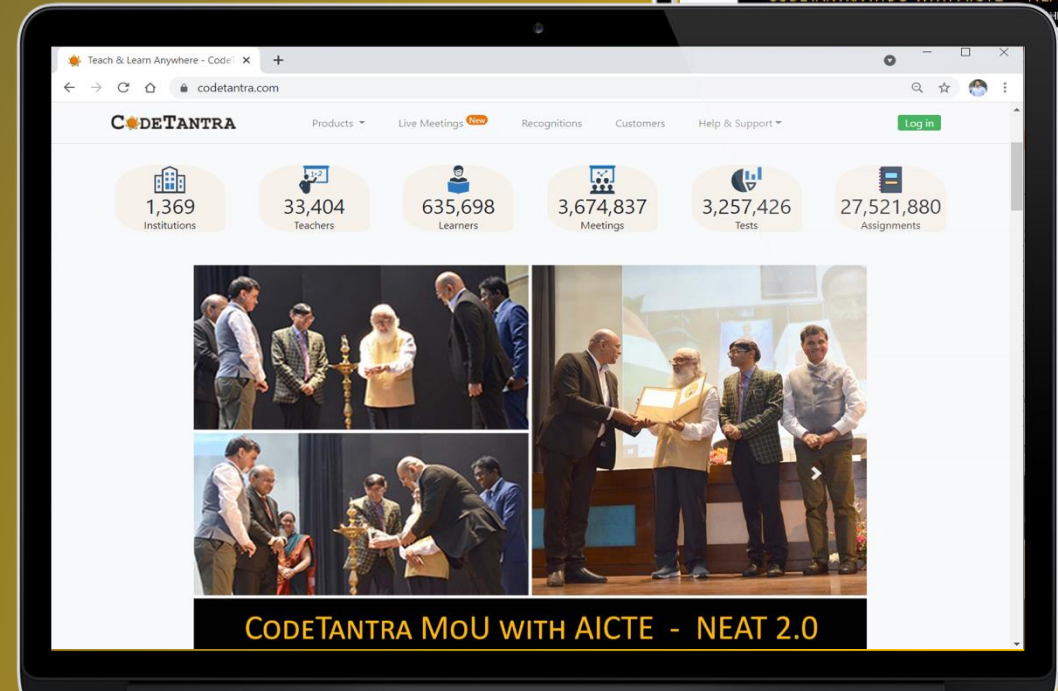
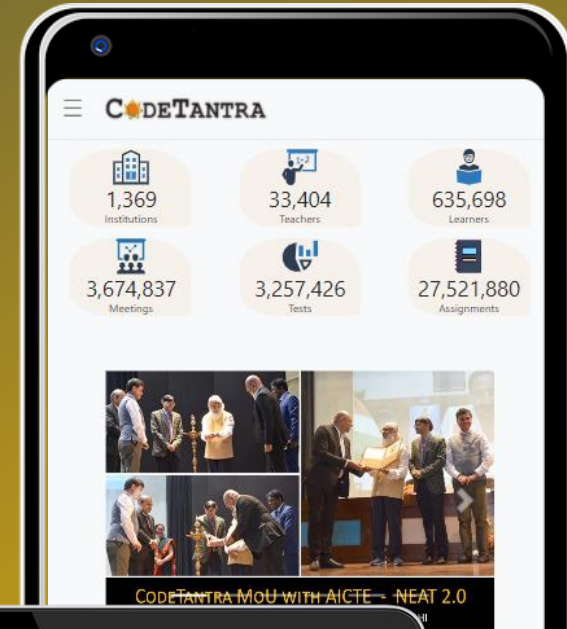




World's most bandwidth-efficient Platform



# How to attempt Test



any device  
t an  
used access,  
ncryption over  
dern browsers  
eded  
s, with video  
k


## Setting up Password

- Automatic email from [support@codetantra.com](mailto:support@codetantra.com) will be sent to users on their registered email ID provided by the institute. The email would contain OTP and a link to set the password to their account.
- Open the email, click on the first link provided in the email. In case, clicking on a link does not work, you can Copy the link and paste it in the browser.
- The OTP is already entered. If it is not entered, please enter the OTP. Enter a secure password and retype the same password to confirm it and click on the submit button

**Congratulations!! Your account is ready.**

The image shows a screenshot of an email and a corresponding web form. The email is from support@codetantra.com, titled "Validation OTP for CodeTantra.com". It contains an OTP of 493316, valid for 7 days, and a link to verify the OTP. The web form, titled "Verify OTP for New Password", has a success message: "The OTP has been sent to your registered email id." It features three input fields: "Enter OTP:" (pre-filled with 493316), "New Password:", and "Confirm Password:". Red arrows point to these fields with instructions: "Enter OTP here, if not pre-filled.", "Enter a new password here.", and "Retype the same password here." There are "Submit" and "Reset" buttons at the bottom.

Validation OTP for CodeTantra.com Inbox x

 **support@codetantra.com** 3:37 AM (0 minutes ago) ☆ ↶ ⋮

to me ▾

Hi  
Please find below your requested OTP:

OTP **493316**  
Please note that this OTP is valid only for **7 days** from the time of requesting.

In case the **OTP expires** you can request it again by clicking on "**Forgot Password**" link.

Click on the below link to enter the OTP given above to set the password.  
Click on this link [https://\[redacted\].codetantra.com/verify-otp-for-password.jsp?id=323884950276015&otp=493316](https://[redacted].codetantra.com/verify-otp-for-password.jsp?id=323884950276015&otp=493316)

After setting the password please use the below link to login.  
[https://\[redacted\].codetantra.com/login.jsp](https://[redacted].codetantra.com/login.jsp)

regards  
Customer Care  
[\[redacted\].codetantra.com](mailto:[redacted].codetantra.com)

**Verify OTP for New Password**

The OTP has been sent to your registered email id. x

Enter OTP:  ← Enter OTP here, if not pre-filled.

New Password :  ← Enter a new password here.

Confirm Password :  ← Retype the same password here.

## Not Received OTP ?

- a) In the case you have not received the mail from [support@codetantra.com](mailto:support@codetantra.com), or E-Mail received but OTP Expired or have forgotten the password, follow the steps below:
  - i. Visit sign-in page (<https://ieeesh-2024.codetantra.com/login.jsp>)
  - ii. Click on **forgot password** button.
  - iii. Enter your registered email id and click on **recover**.
  - iv. You would receive an email with OTP from [support@codetantra.com](mailto:support@codetantra.com).
  - v. Follow the instructions in the mail to reset the password by clicking the 1st link.

### Important Notes

- ✓ The OTP generated is valid for only 10 minutes.
- ✓ If you encounter an error during login, then ensure that you are on the correct login URL. If this error occurs even on a correct login URL, cross check with your institute to see if you are using the correct email address registered on CodeTantra platform.

IEEE Region10 IEEE INDIAN COUNCIL  
SDA Committee of India Council

EMAIL

PASSWORD

LOGIN

**Forgot your password?**  
Not your institution?

IEEE Region10 IEEE INDIAN COUNCIL  
SDA Committee of India Council

Recover Password

EMAIL

**Recover**

Remember your password? Sign in







# Device Requirement



Laptop/Desktop

Open Test

# Which combination of OS and browser is supported?

For Laptop/Desktop			
Operating System	 Windows	 Mac	 Linux
Supported Browser	 CodeTantra Secure Exam Application (SEA)	 CodeTantra Secure Exam Application (SEA)	 CodeTantra Secure Exam Application (SEA)

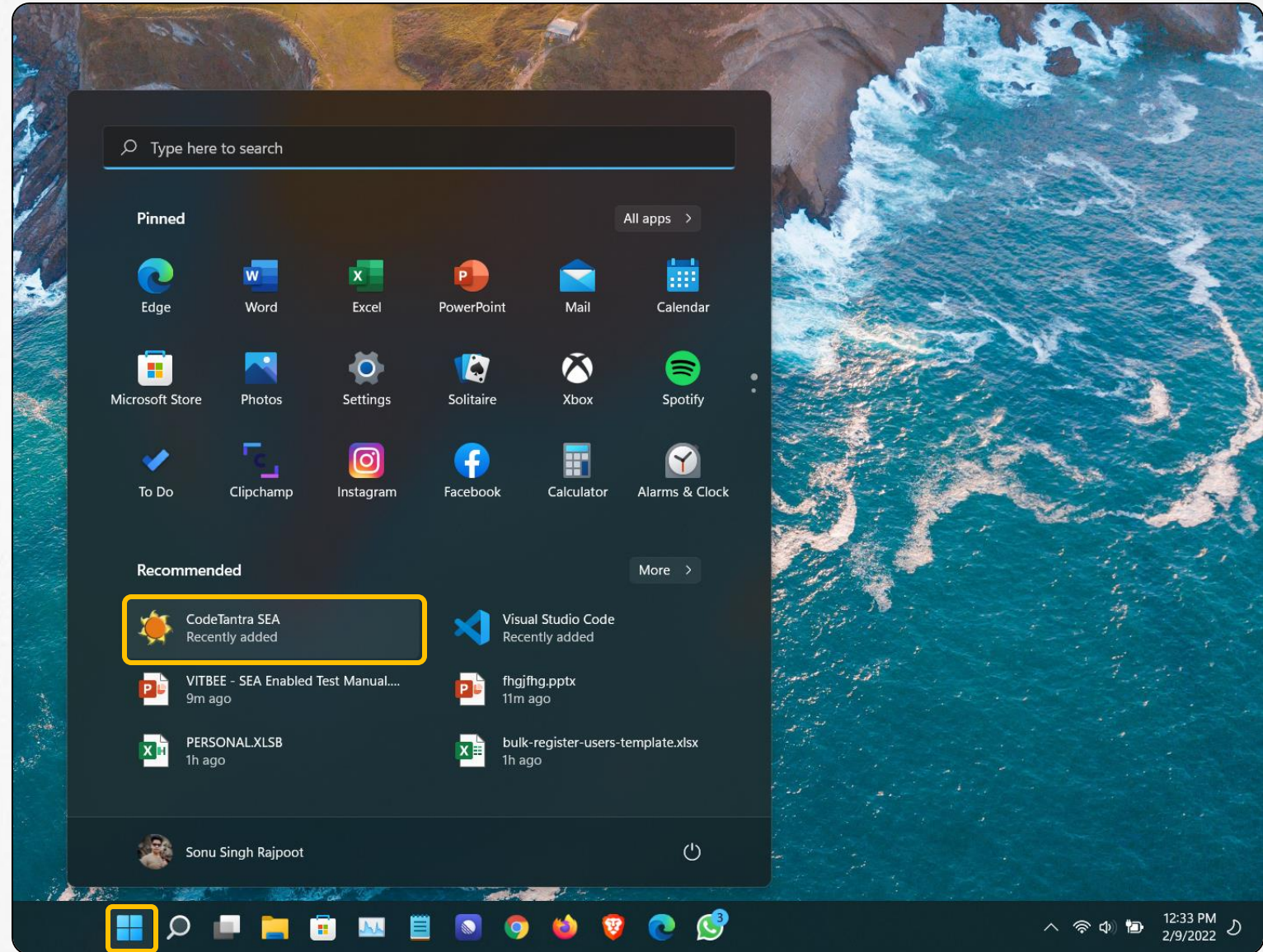


## Step 01 : Run the SEA

- i. Go to Start Menu
- ii. Find and click on CodeTantra SEA

### Important Notes

- ✓ If CodeTantra SEA is **not** installed, then while attempting the mock test you will be redirected to download one.
- ✓ In case of **invalid Login id / Password** error, ensure that the logo on the login page is same as that of your institute. If not, then you are on wrong login page. Recheck the login link.



## Step 02 : Search Your Institute

- Click on the drop down and type [ieeessh-2024.codetantra.com](http://ieeessh-2024.codetantra.com) to fetch your institute from the list.



## Step 03 : Select Your Institution

Click on [ieeesh-2024.codetantra.com](http://ieeesh-2024.codetantra.com) button from the suggested list and click [Go](#).





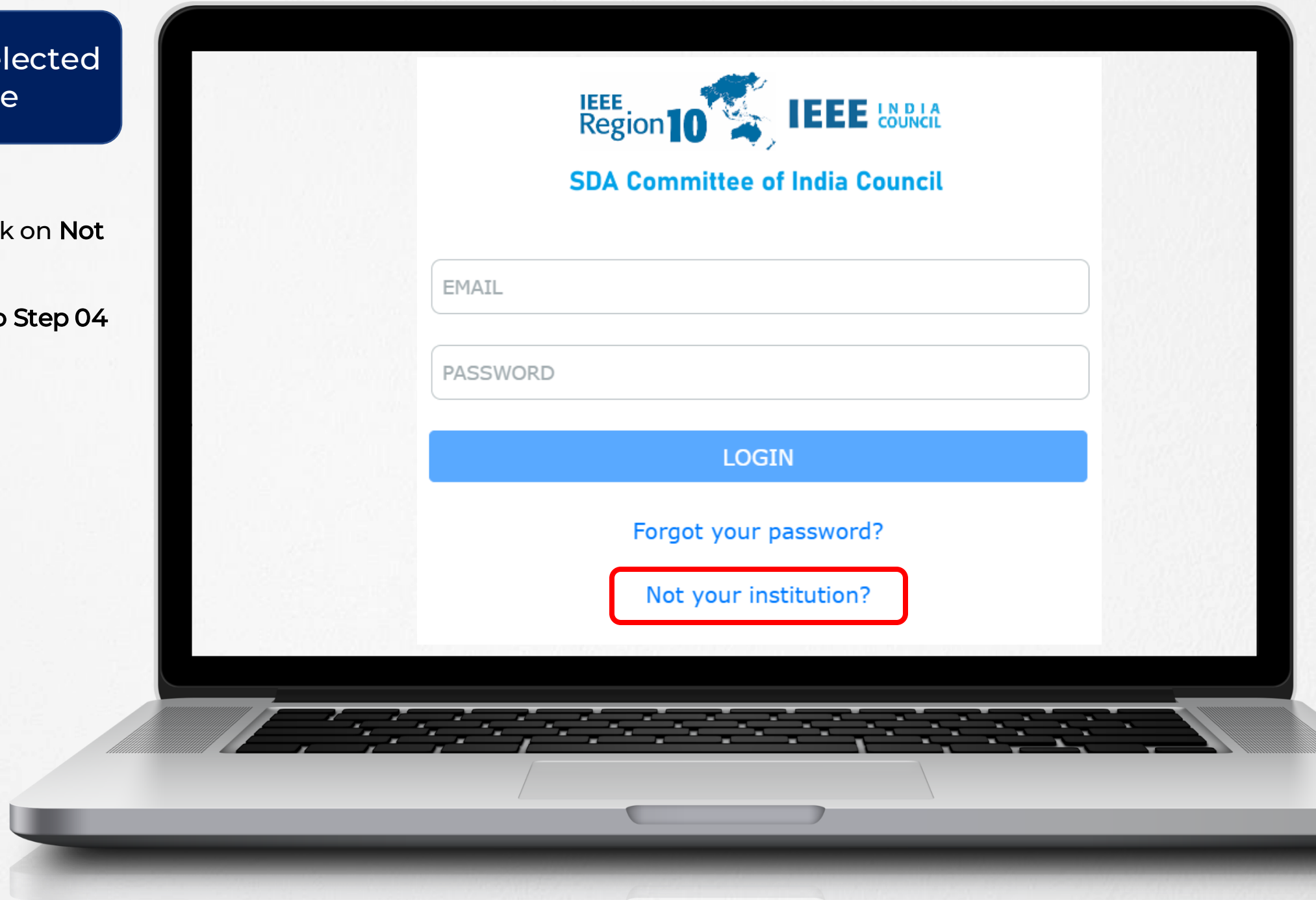
## Step 04 : Login

Enter Your Email Id and Password in the respective field and then click on Login button.



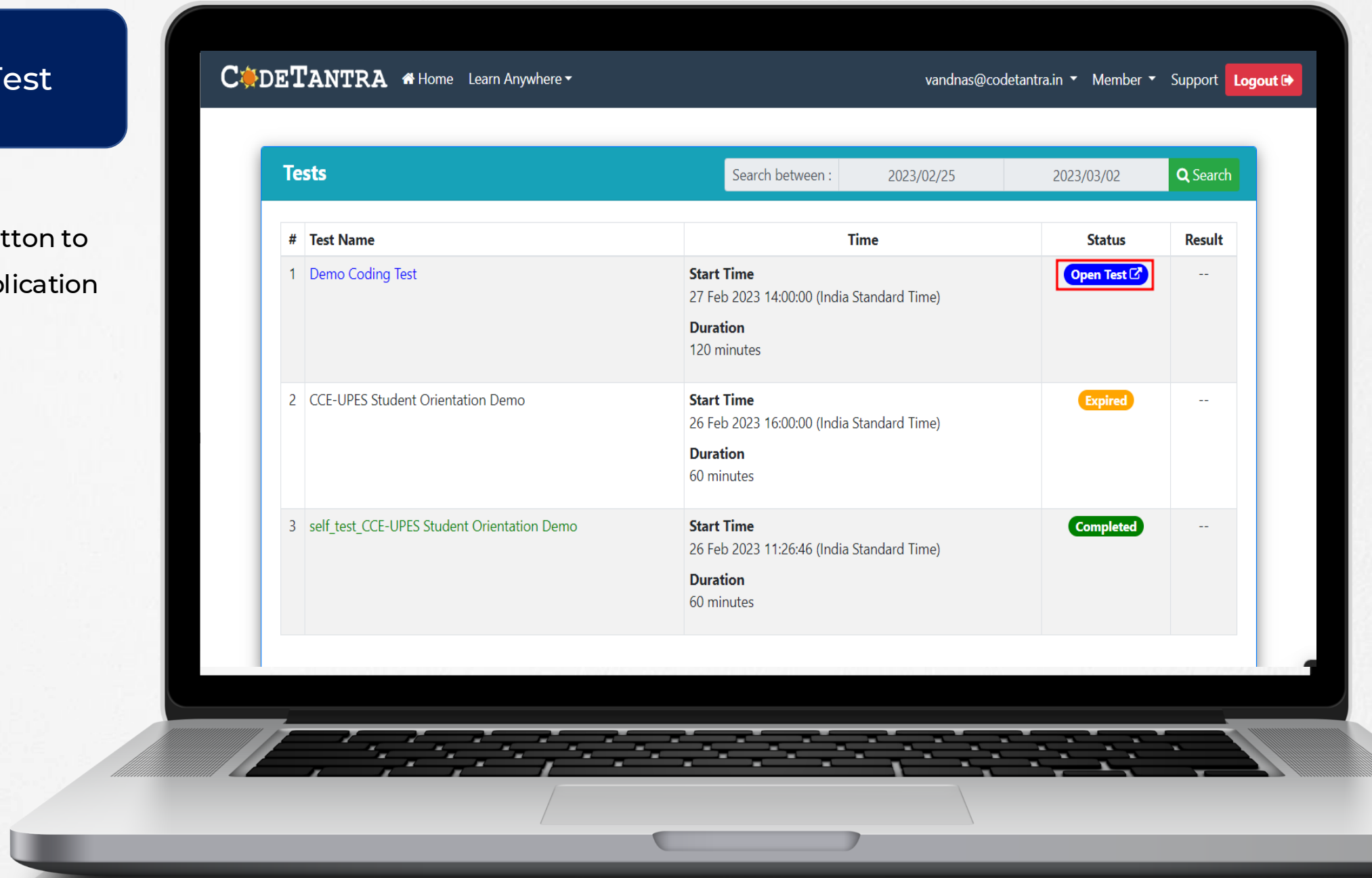
## What if you have selected wrong institute

- On the login page, click on **Not Your Institution?**
- Again follow **Step 02** to **Step 04**



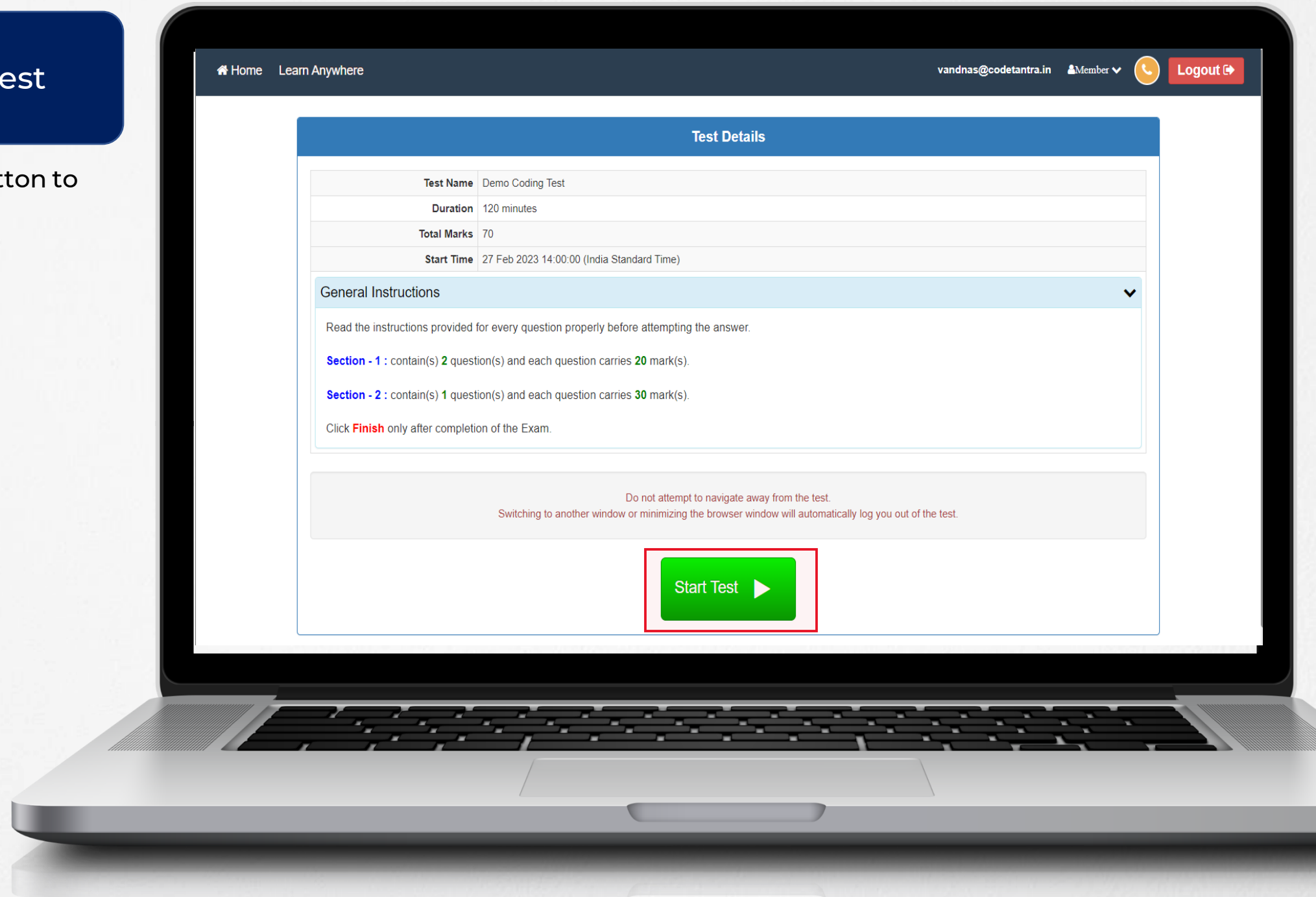
## Step 05 : Open Test

Click on the **Open Test** button to initiate the test in the application window.



## Step 06 : Start Test

Click on the **Start Test** button to start the test.



## Step 07 : Answer the coding questions

Type the code in the Code Editor and click on **Submit** button to check if all test cases have passed or not.

The screenshot displays a coding platform interface. On the left, the problem statement for 'UP-DOWN/ZIG-ZAG ARRAY' is shown. It asks to transform an array of N integers into a zig-zag pattern (UP-DOWN) by replacing elements with any integer, minimizing the number of replacements. The function signature is `int minimumChanges(int Arr[], int ArrLen)`. Constraints are  $1 \leq N \leq 10^6$  and  $1 \leq \text{ARR}[i] \leq 10^9$ . On the right, the code editor shows a C program for 'CTC45.c' with the following code:

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4
5 int minimumChanges(int Arr[], int ArrLen) {
6     // Write code here
7 }
8
9
10 int readIntArray(char *argsArray, int arr[]) {
11     int col = 0;
12     char *token = strtok(argsArray, " ");
13     while (token != NULL) {
14         arr[col] = atoi(token);
15         token = strtok(NULL, " ");
16         col++;
17     }
18     return col;
19 }
20
21 int main(int argc, char *argv[]) {
22     int Arr[strlen(argv[1])];
23     int ArrLen = readIntArray(argv[1], Arr);
24     printf("%d\n", minimumChanges(Arr, ArrLen));
25 }
```

The terminal shows the execution of the program, which successfully runs and outputs the result. The 'Submit' button is highlighted in red.



+  
• ○ Let's explore the other  
○ elements of the  
Interface

---

Previous Button



Next Button



Alternate Navigation Button

You can use alternate Previous and Next button to navigate through questions in addition to traditional Prev and Next Button.

Section 1 of 2 Q1 of 2 Marks 20 Jump to General Instructions 110%+ Test time left: 01:59:16

Write the code

**Problem Statement:**  
**UP-DOWN/ZIG-ZAG ARRAY**

You are given an array/list **Arr** containing **N** integers, change it in such a way that it follows a **UP-DOWN** that is also called zig-zag pattern. A zig-zag array is one where for each integer, its adjacent integers are both greater than or less than itself.

In other words, using **D** to mean a lower/down value and **U** to mean higher/up, the array follows either the pattern **[D,U,D,U...]** or **[U,D,U,D...]**. To make the array a **UP-DOWN/zig-zag** array.

You can replace any element with any other integer (positive, negative, or zero) and find out the minimum number of replacements required to accomplish this task.

Complete the function **minimumChanges** with the following parameter:

- integer **Arr[N]**: an array/list of N integers

Function will return

- integer: The minimum number of changes required to turn array/list Arr into a UP-DOWN/zig-zag array

**Note:** Coders must strictly follow the function rules given below, So that you will be well aware of the parameters passed to the function and also how to return results from the required function.

Constraints to be followed :

- $1 \leq N \leq 10^5$
- $1 \leq ARR[i] \leq 10^9$

Finish Clear

Correct/complete the code. The code highlighted in   is non-editable.

CTC45.c C Submit

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4
5 int minimumChanges(int Arr[], int ArrLen) {
6     // Write code here
7
8 }
9
10 int readIntArray(char *argsArray, int arr[]) {
11     int col = 0;
12     char *token = strtok(argsArray, ",");
13     while (token != NULL) {
14         arr[col] = atoi(token);
15         token = strtok(NULL, ",");
16         col++;
17     }
18     return col;
19 }
20 int main(int argc, char *argv[]) {
21     int Arr[strlen(argv[1])];
22     int ArrLen = readIntArray(argv[1], Arr);
23     printf("%d\n", minimumChanges(Arr, ArrLen));
24 }
```

Terminal Execution Results

```
>_
code@tantra:~$ cd $HOME && mkdir -p ct-multi-work/multiLanguage && cd ct-multi-work/multiLanguage
code@tantra:~/multiLanguage$ cd $HOME && mkdir -p ct-multi-work/multiLanguage && cd ct-multi-work/multiLanguage
code@tantra:~/multiLanguage$ cd $HOME && mkdir -p ct-multi-work/multiLanguage && cd ct-multi-work/multiLanguage
code@tantra:~/multiLanguage$ cd $HOME && mkdir -p ct-multi-work/multiLanguage && cd ct-multi-work/multiLanguage
code@tantra:~/multiLanguage$
```

Submit Prev Next

Traditional Navigation Button

## Using the Jump to navigation panel to navigate

You can access any question of any section from the current question by clicking on the desired question number.

Please note, if **Jump To** button is not visible then most likely it has been disabled as per the institute guidelines

The screenshot displays a coding platform interface. At the top, a navigation panel shows 'Section 1 of 2', 'Q1 of 2', and 'Marks 20'. A yellow overlay highlights two sections: 'Section 1 20 mark(s)' with a '1' icon and 'Section 2 30 mark(s)' with a '1' icon. Below this, a legend indicates '2 - Not Visited', '1 - Visited', and '0 - Attempted'. The main area contains a 'Problem Statement' for a function named `balancing_array`. The problem involves finding the index of the smallest pivot element in an array. The code editor shows a C program with a `readLongArray` function and a `main` function. The terminal shows the command `cd $HOME && mkdir -p ct-multi-work/multiLanguage && cd ct-multi-work/multiLanguage` being executed. At the bottom, there are buttons for 'Finish', 'Clear', 'Submit', 'Prev', and 'Next'. A 'Test time left: 01:57:40' indicator is visible in the top right corner.

## Viewing the General Instructions

- Click on **General Instructions** button to open instruction panel.
- Click on close after viewing the Instructions

The screenshot displays a coding platform interface. At the top, there are navigation elements: 'Section 1 of 2', 'Q1 of 2', 'Marks 20', and a 'Jump to' button. A 'General Instructions' button is highlighted with a red box in the top right corner. Below this, the main area is divided into a 'Problem Statement' section on the left and a code editor on the right. The 'Problem Statement' section contains text about finding the index of the smallest pivot element in an array. The code editor shows a file named 'CTC26.c' with two lines of C code: `#include <stdio.h>` and `#include <stdlib.h>`. A modal dialog titled 'General Instructions' is overlaid on the interface. It contains the following text: 'Read the instructions provided for every question properly before attempting the answer.', 'Section - 1 : contain(s) 2 question(s) and each question carries 20 mark(s).', 'Section - 2 : contain(s) 1 question(s) and each question carries 30 mark(s).', and 'Click **Finish** only after completion of the Exam.' A 'Close' button is located at the bottom right of the modal. At the bottom of the interface, there are buttons for 'Finish', 'Clear', 'Submit', 'Prev', and 'Next'. The 'Terminal' and 'Execution Results' sections are also visible at the bottom right.

## Zooming in/out

You can resize the text of the question and answers by clicking on **-** and **+** button.

If there is any image in the question or option, you can click on the image to zoom in. After viewing the image you can click on minimize button available at top right corner of the image.

Section 1 of 2 Q1 of 2 Marks 20 Jump to General Instructions 70% Test time left: 01:54:38

Write the code

**Problem Statement:**  
You are having **N** no of elements in an array/list name **Arr** , find the index of the smallest pivot element in the array for which the sum of all elements to the left and to the right are equal.

**Note:** The array cannot be reordered and indexing in the array starts from 0.

You have to complete function **balancing\_array** with parameter's passed:

**integer Arr[N]:** Integers array/list of size N

Functions return value is :

Integer: Representing the index of the pivot element.

**Note:** Coders must strictly follow the function rules given below, So that you will be well aware of the parameters passed to the function and also how to return results from the required function.

**Note :** Indexing in the array/list starts from 0.

**Constraints to be followed :**

$3 \leq N \leq 10^5$   
 $1 \leq Arr[i] \leq 10^4$ , where  $0 \leq i < N$

**Sample test case is :**

**Input :**

Correct/complete the code. The code highlighted in ■ is non-editable.

CTC26.c C Submit

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4
5 int balancing_array(long Arr[], int ArrLen) {
6     // Write code here
7
8 }
9
10 int readLongArray(char *argsArray, long arr[]) {
11     int col = 0;
12     char *token = strtok(argsArray, ",");
13     while (token != NULL) {
14         arr[col] = atol(token);
15         token = strtok(NULL, ",");
16         col++;
17     }
18     return col;
19 }
20 int main(int argc, char *argv[]) {
21     long Arr[strlen(argv[1])];
22     int ArrLen = readLongArray(argv[1], Arr);
23     printf("%d\n", balancing_array(Arr, ArrLen));
```

Terminal Execution Results

```
>_
code@tantra:~$ cd $HOME && mkdir -p ct-multi-work/multiLanguage && cd ct-multi-work/multiLanguage
code@tantra:~$
```

Finish Clear Submit Prev Next



## Step 08 : Finishing the test

- i. On the last question of the test, **Next** button is disabled.
- ii. To finish the test, click on the **Finish** Button.
- iii. A Test Summary will appear, after verifying the Summary, you can click on **Yes, I want to Finish and Exit.**

Section 1 of 2 Q1 of 2 Marks 20 Jump to General Instructions 70% Test time left: 01:51:43

Write the code

**Problem Statement:**  
You are having **N** no of elements in an array. You have to find the sum of all elements to the left and right of each element.  
**Note:** The array cannot be reordered.  
You have to complete function `balanceArray` in the given code snippet.

`integer Arr[N]:` Integers array/list of size N.

Functions return value is :  
Integer: Representing the index of the element.

**Note:** Coders must strictly follow the constraints and also how to return results from the required function.

**Note :** Indexing in the array/list starts from 0.

**Constraints to be followed :**  
 $3 \leq N \leq 10^5$   
 $1 \leq Arr[i] \leq 10^4$ , where  $0 \leq i < N$

**Sample test case is :**  
**Input :**  
1 2 3 4 6 // Array/list Arr  
**Output :**

**Test Summary**

Sections	No. of Questions	Answered	Not Answered	Not visited
Section 1	2	0	1	1
Section 2	1	0	0	1

**There is still time for the Test to finish. Do you still want to exit?**

**No, I want to continue the Test.** **Yes, I want to Finish and Exit.**

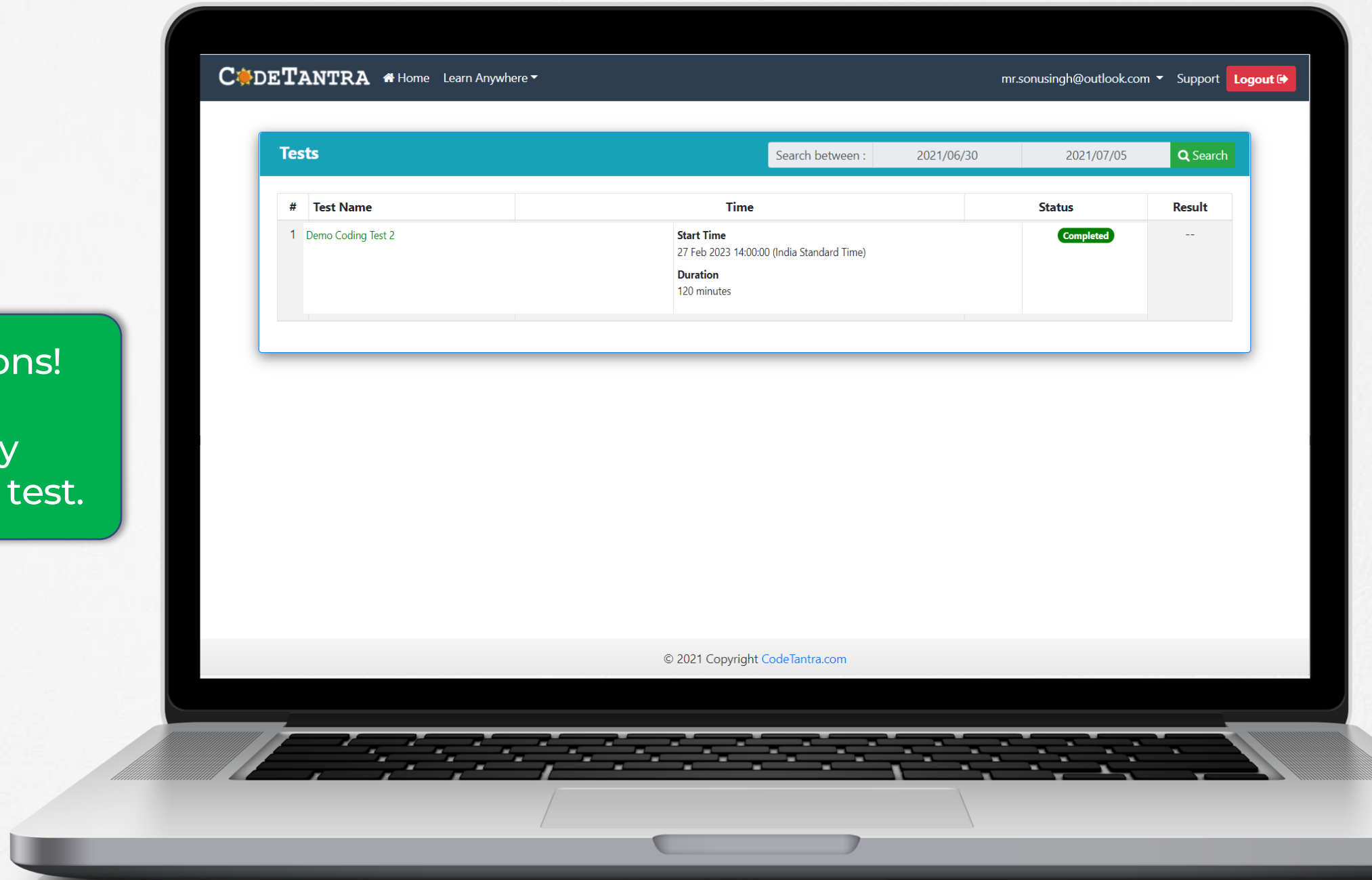
```
18 return col;
19 }
20 int main(int argc, char *argv[]) {
21     long Arr[strlen(argv[1])];
22     int ArrLen = readLongArray(argv[1], Arr);
23     printf("%d\n", balancing_array(Arr, ArrLen));
24     return 0;
}
```

**Terminal** **Execution Results**

```
code@tantra:~$ cd $HOME && mkdir -p ct-multi-work/multiLanguage && cd ct-multi-work/multiLanguage
code@tantra:~$
```

**Finish** **Clear** **Submit** **Prev** **Next**

Congratulations!  
You have  
successfully  
submitted the test.





## For any technical query

- [support@codetantra.com](mailto:support@codetantra.com)
- +91 799 541 7777
- +91 630 500 7479
- +91 630 500 8997