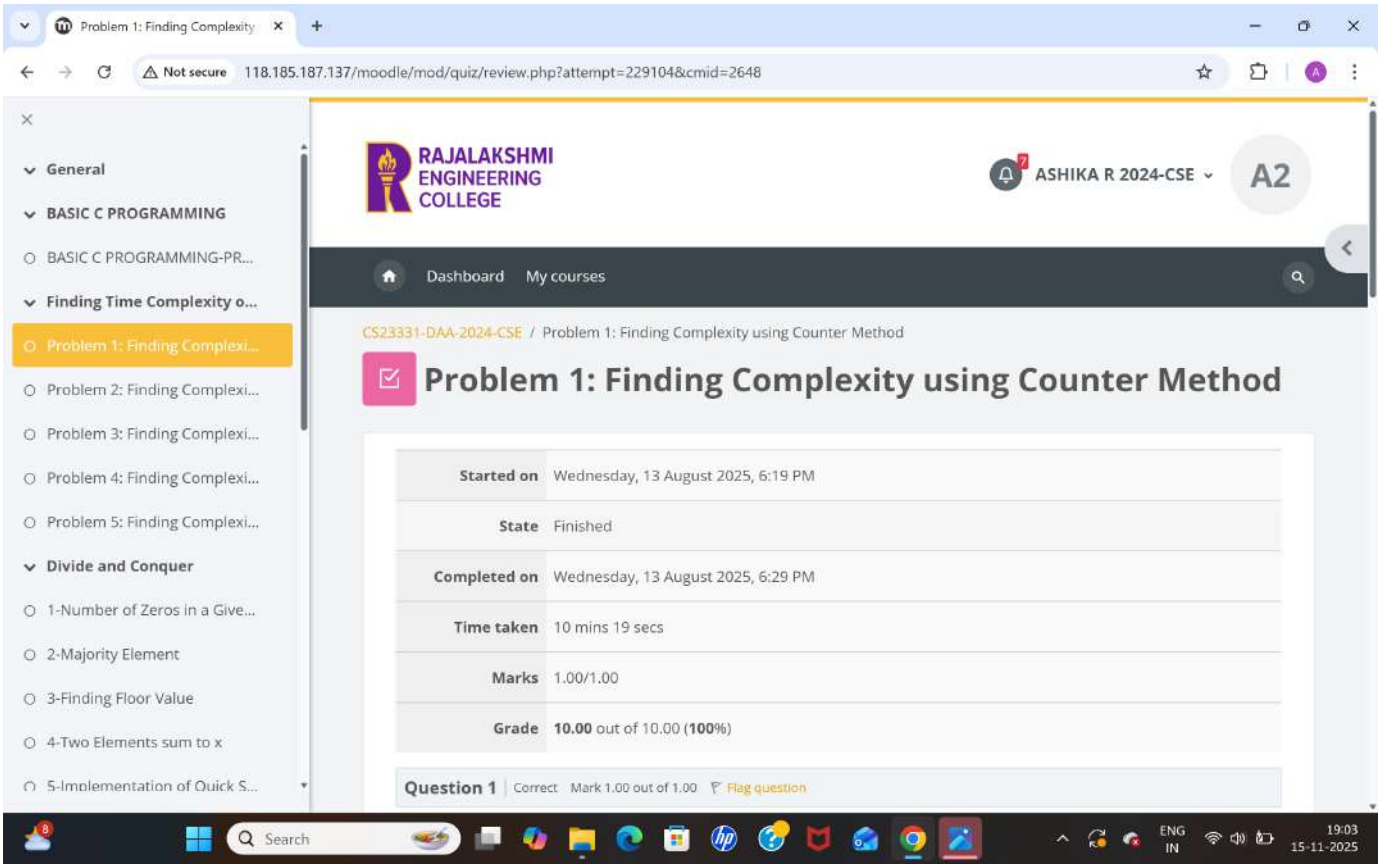
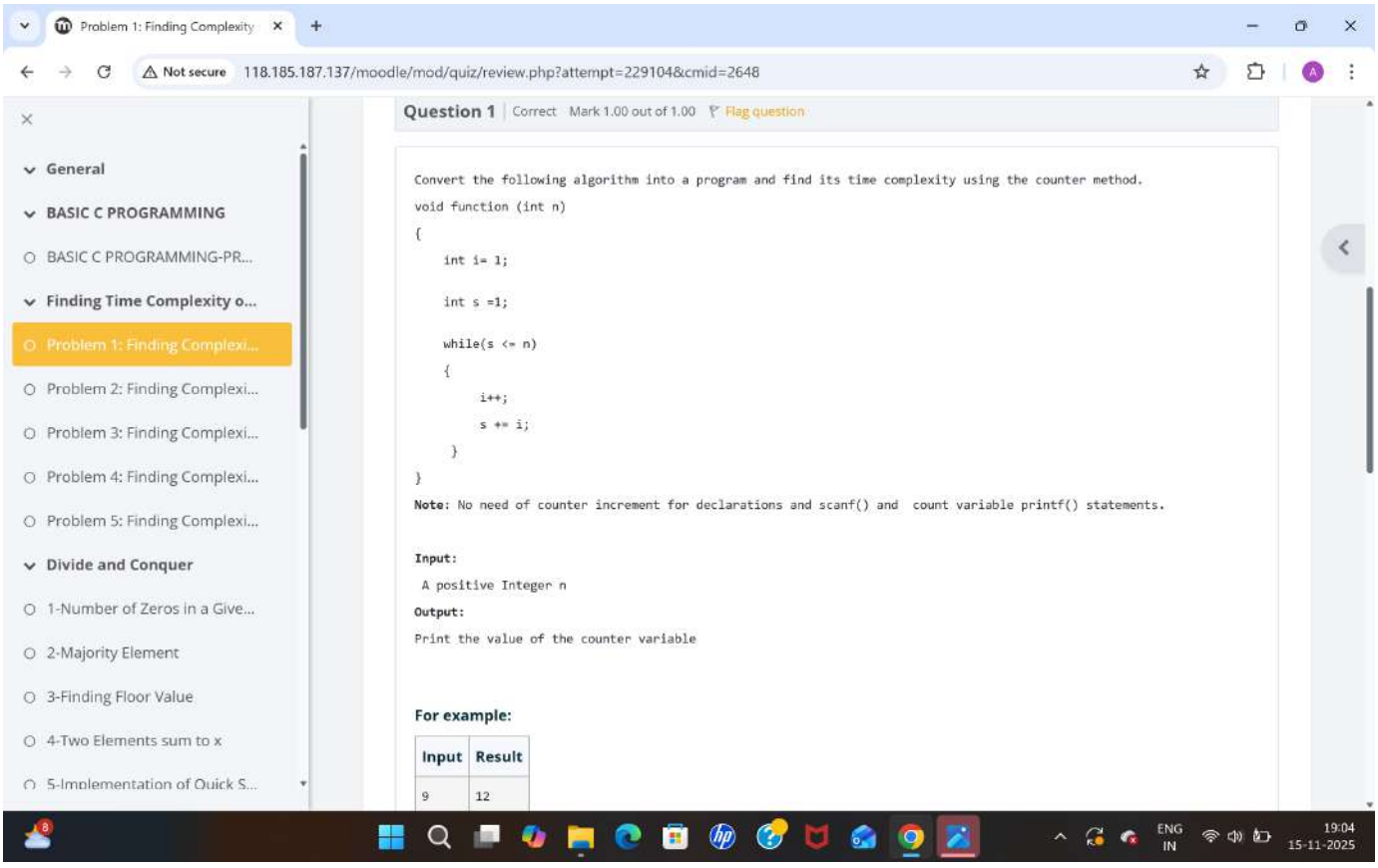
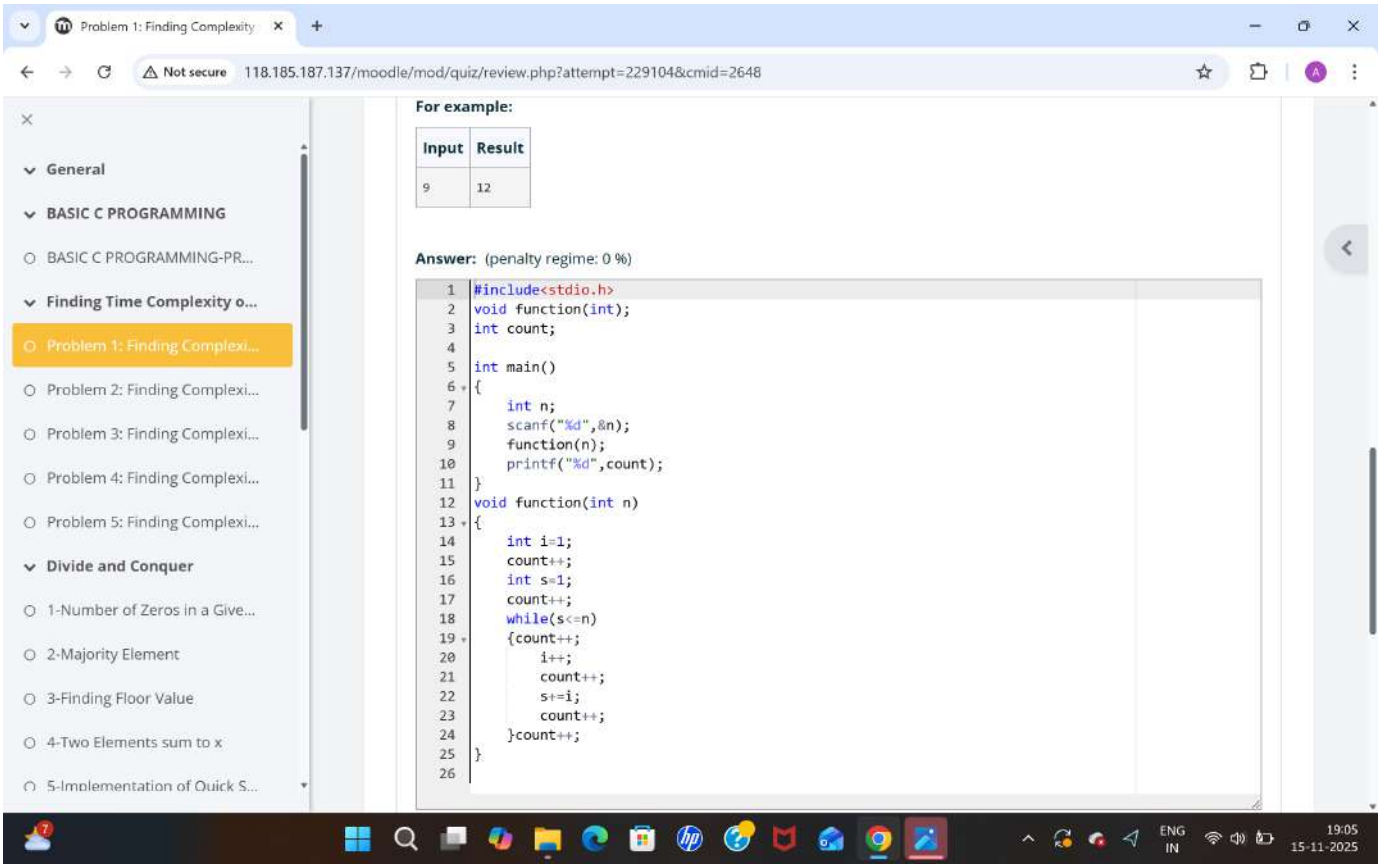
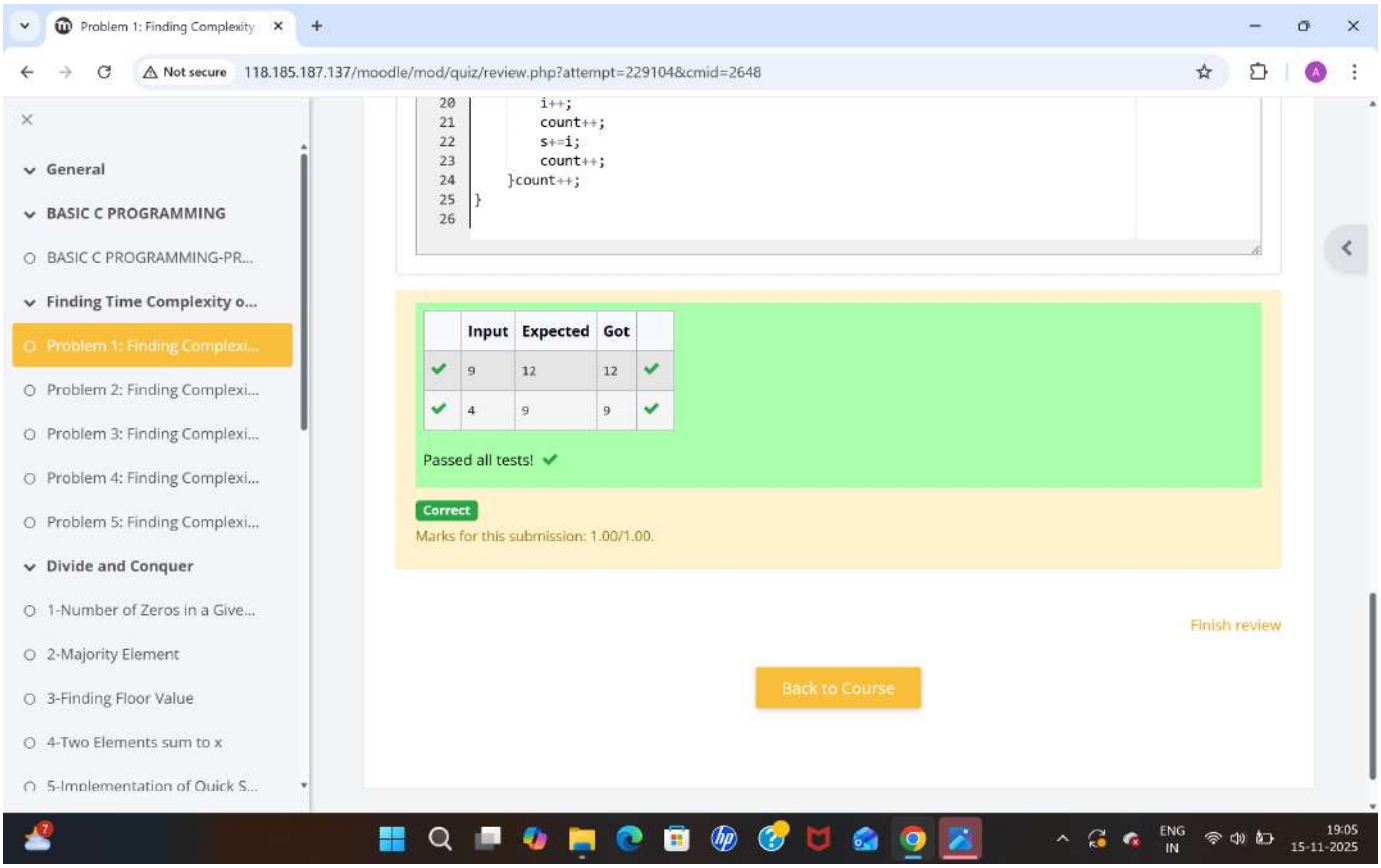
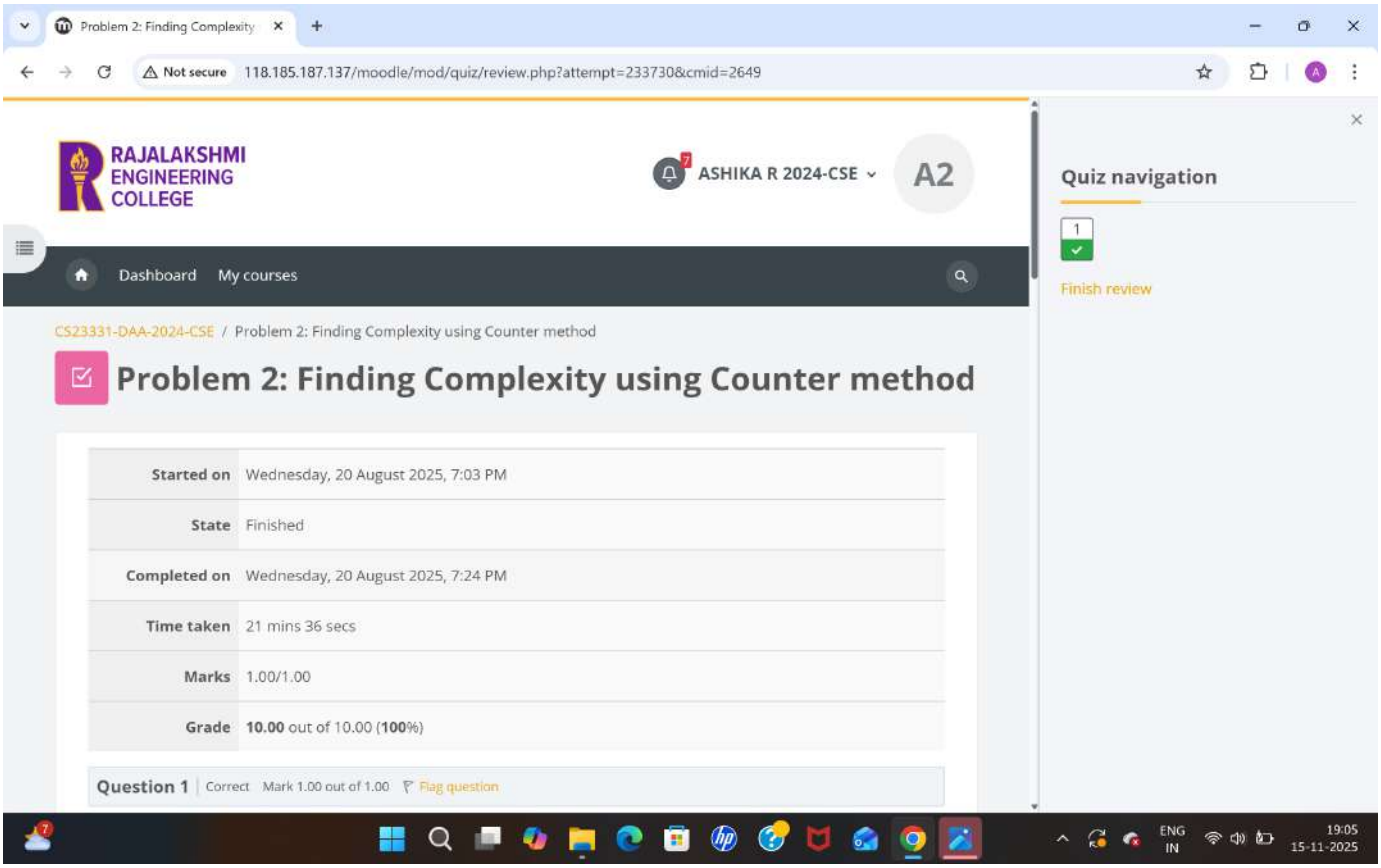
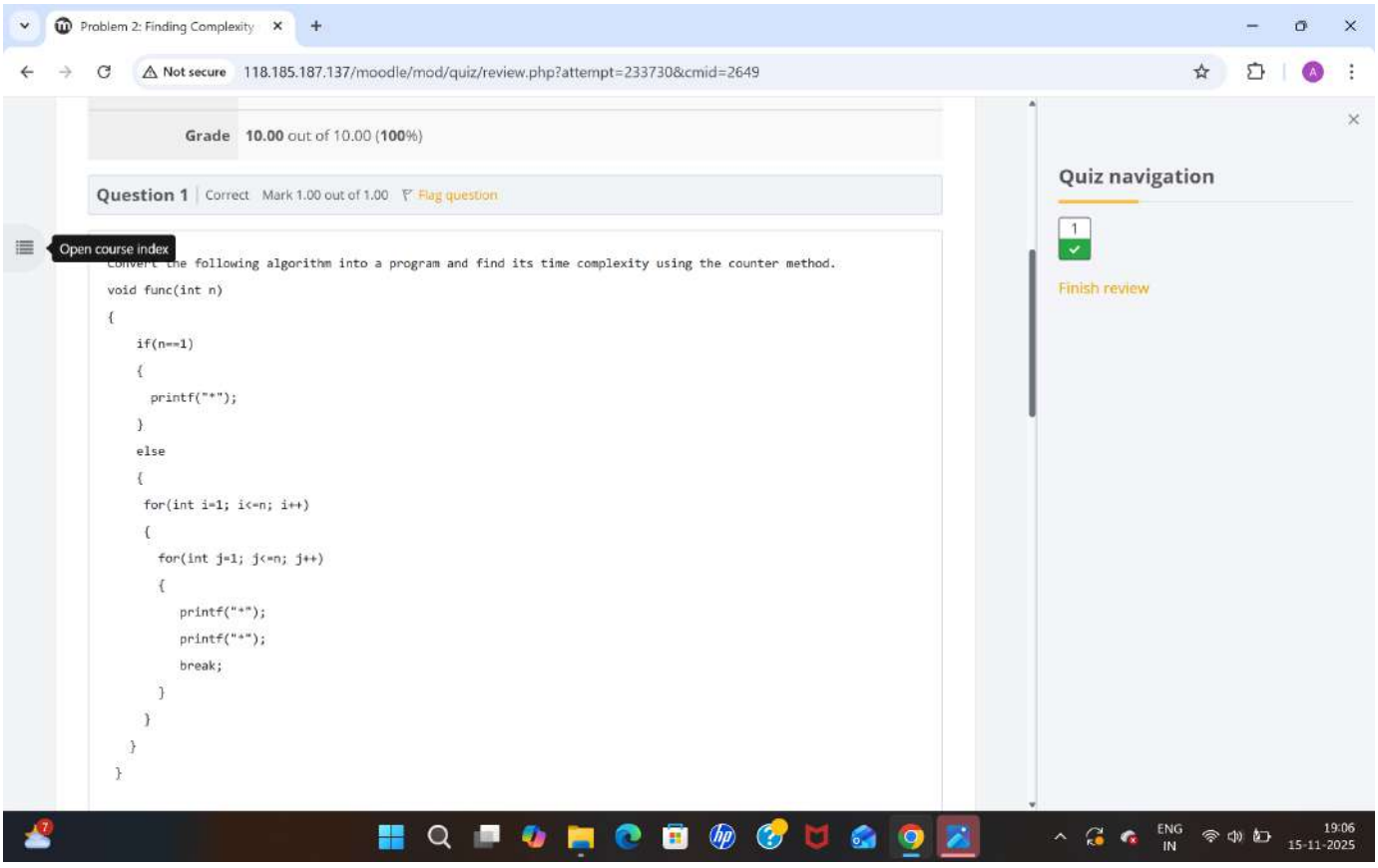

Problem 1: Finding Complexity
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=229104&cmid=2648
General
BASIC C PROGRAMMING
BASIC C PROGRAMMING-PR...
Finding Time Complexity o...
Problem 1: Finding Complexi...
Problem 2: Finding Complexi...
Problem 3: Finding Complexi...
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / Problem 1: Finding Complexity using Counter Method
Problem 1: Finding Complexity using Counter Method
Started on Wednesday, 13 August 2025, 6:19 PM
State Finished
Completed on Wednesday, 13 August 2025, 6:29 PM
Time taken 10 mins 19 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Search
ENG IN
19:03
15-11-2025

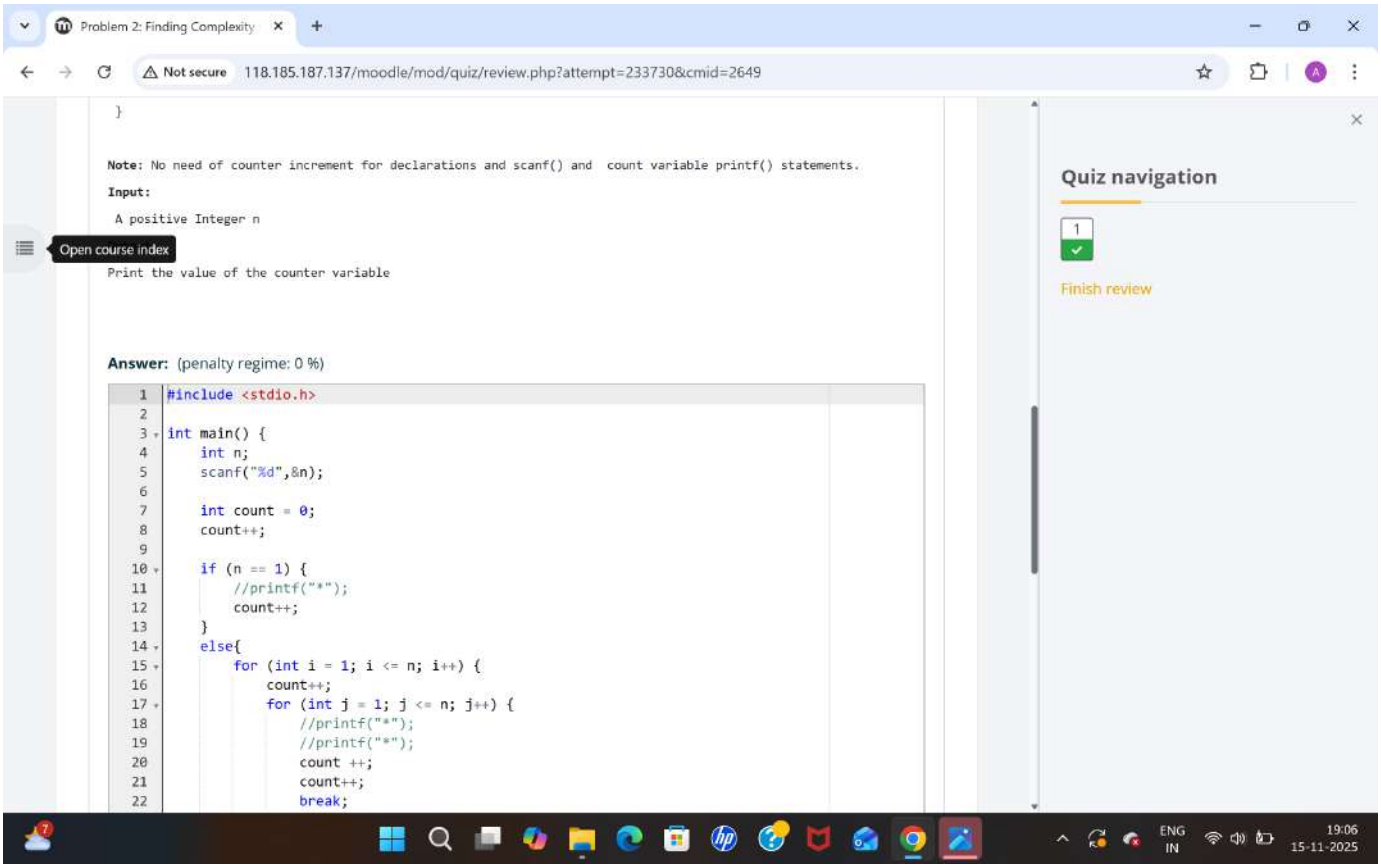

Problem 1: Finding Complexity
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=229104&cmid=2648
General
BASIC C PROGRAMMING
BASIC C PROGRAMMING-PR...
Finding Time Complexity o...
Problem 1: Finding Complexi...
Problem 2: Finding Complexi...
Problem 3: Finding Complexi...
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
Question 1
Correct
Mark 1.00 out of 1.00
Flag question
Convert the following algorithm into a program and find its time complexity using the counter method.
void function (int n)
{
int i= 1;
int s =1;
while(s <= n)
{
i++;
s += i;
}
}
Note: No need of counter increment for declarations and scanf() and count variable printf() statements.
Input:
A positive Integer n
Output:
Print the value of the counter variable
For example:
Input
Result
9
12
ENG
IN
19:04
15-11-2025

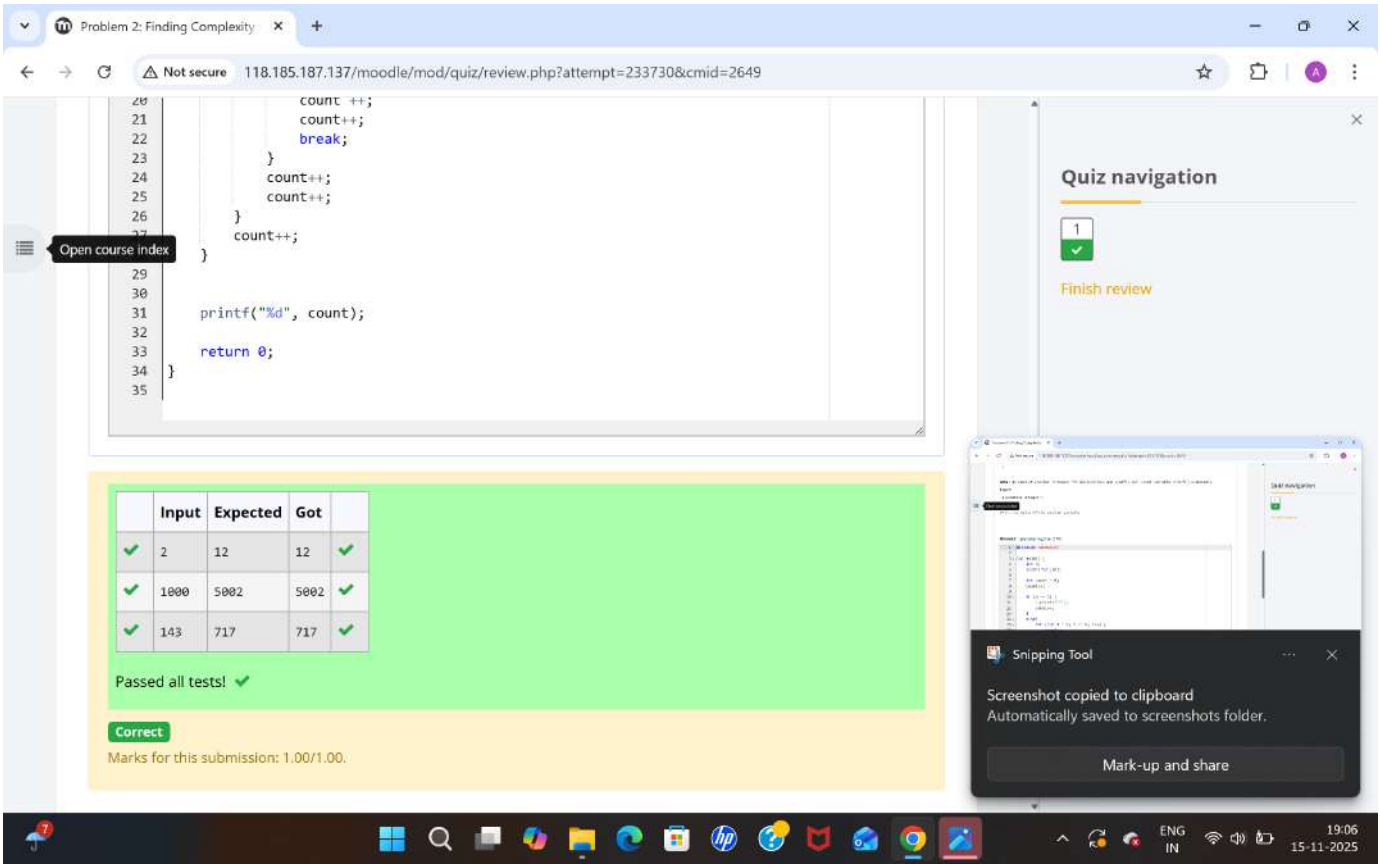

Problem 1: Finding Complexity
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=229104&cmid=2648
General
BASIC C PROGRAMMING
BASIC C PROGRAMMING-PR...
Finding Time Complexity o...
Problem 1: Finding Complexi...
Problem 2: Finding Complexi...
Problem 3: Finding Complexi...
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
For example:
Input
Result
9
12
Answer: (penalty regime: 0 %)
#include<stdio.h>
void function(int);
int count;
int main()
{
int n;
scanf("%d",&n);
function(n);
printf("%d",count);
}
void function(int n)
{
int i=1;
count++;
int s=1;
count++;
while(s<=n)
{count++;
i++;
count++;
s+=i;
count++;
}count++;
}
ENG
IN
19:05
15-11-2025

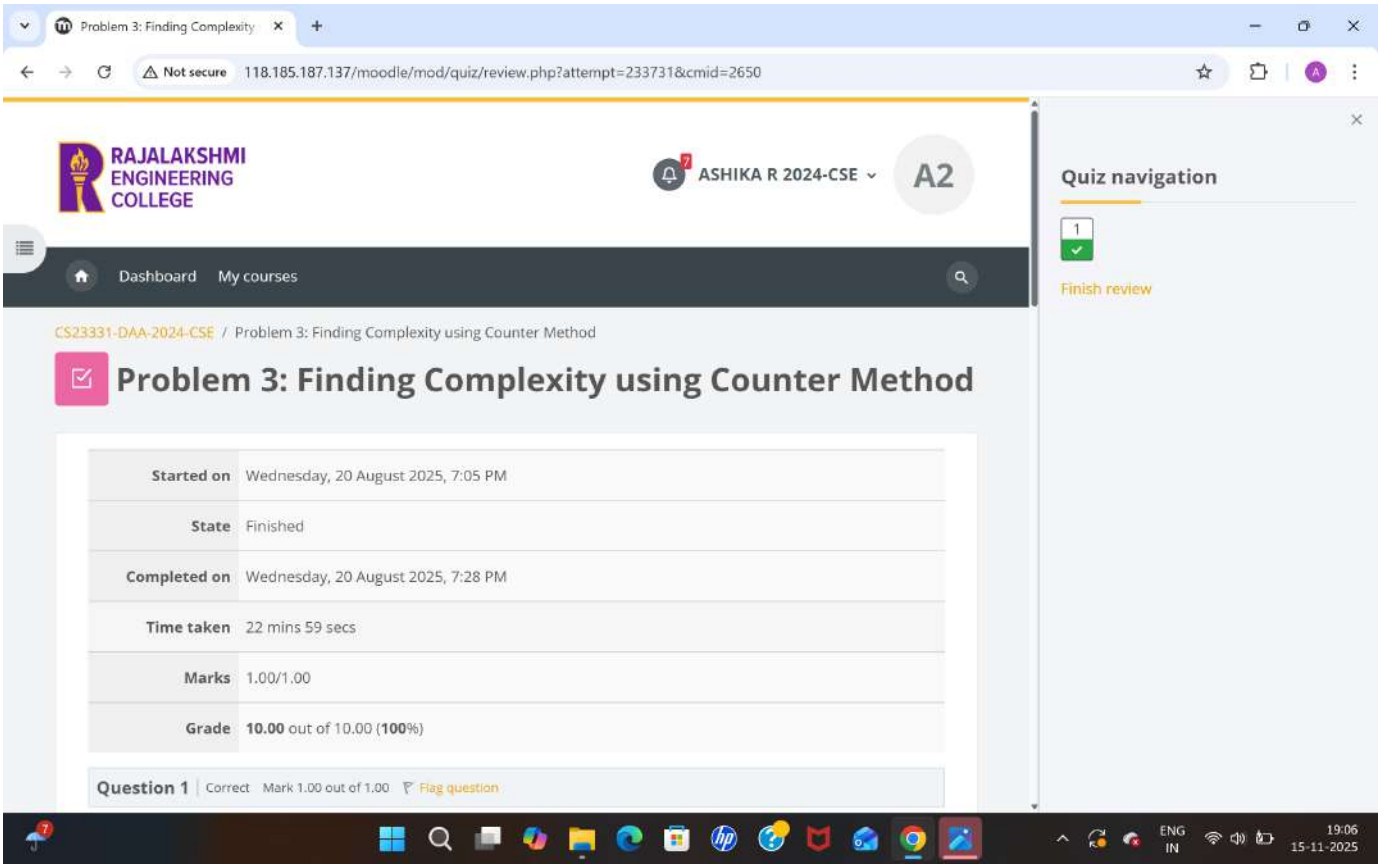

Problem 1: Finding Complexity
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=229104&cmid=2648
i++;
count++;
s+=i;
count++;
}count++;
}
General
BASIC C PROGRAMMING
BASIC C PROGRAMMING-PR...
Finding Time Complexity o...
Problem 1: Finding Complexi...
Problem 2: Finding Complexi...
Problem 3: Finding Complexi...
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
Input
Expected
Got
9
12
12
4
9
9
Passed all tests!
Correct
Marks for this submission: 1.00/1.00.
Finish review
Back to Course
ENG
IN
19:05
15-11-2025


Problem 2: Finding Complexity
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233730&cmid=2649
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / Problem 2: Finding Complexity using Counter method
Problem 2: Finding Complexity using Counter method
Started on Wednesday, 20 August 2025, 7:03 PM
State Finished
Completed on Wednesday, 20 August 2025, 7:24 PM
Time taken 21 mins 36 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
ENG IN
19:05
15-11-2025


Problem 2: Finding Complexity
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233730&cmid=2649
Grade 10.00 out of 10.00 (100%)
Question 1
Correct
Mark 1.00 out of 1.00
Flag question
Open course index
Convert the following algorithm into a program and find its time complexity using the counter method.
void func(int n)
{
if(n==1)
{
printf("*");
}
else
{
for(int i=1; i<=n; i++)
{
for(int j=1; j<=n; j++)
{
printf("*");
printf("*");
break;
}
}
}
}
Quiz navigation
1
Finish review
ENG
IN
19:06
15-11-2025


Problem 2: Finding Complexity
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233730&cmid=2649
}
Note: No need of counter increment for declarations and scanf() and count variable printf() statements.
Input:
A positive Integer n
Open course index
Print the value of the counter variable
Answer: (penalty regime: 0 %)
#include <stdio.h>
int main() {
int n;
scanf("%d",&n);
int count = 0;
count++;
if (n == 1) {
//printf("*");
count++;
}
else{
for (int i = 1; i <= n; i++) {
count++;
for (int j = 1; j <= n; j++) {
//printf("*");
//printf("*");
count ++;
count++;
break;
Quiz navigation
1
Finish review
ENG
IN
19:06
15-11-2025


Problem 2: Finding Complexity
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233730&cmid=2649
count ++;
count++;
break;
}
count++;
count++;
}
count++;
}
printf("%d", count);
return 0;
}
Open course index
Quiz navigation
1
Finish review
Input
Expected
Got
2
12
12
1000
5002
5002
143
717
717
Passed all tests!
Correct
Marks for this submission: 1.00/1.00.
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:06
15-11-2025


Problem 3: Finding Complexity
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233731&cmid=2650
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / Problem 3: Finding Complexity using Counter Method
Problem 3: Finding Complexity using Counter Method
Started on Wednesday, 20 August 2025, 7:05 PM
State Finished
Completed on Wednesday, 20 August 2025, 7:28 PM
Time taken 22 mins 59 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
ENG IN
19:06
15-11-2025

Problem 3: Finding Complexity

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=233731&cmid=2650

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

Convert the following algorithm into a program and find its time complexity using counter method.

Open course index

```
m) {
 {
    for (i = 1; i <= num;++i)
    {
     if (num % i== 0)
        {
          printf("%d ", i);
        }
     }
 }
```

Note: No need of counter increment for declarations and scanf() and counter variable printf() statement.

Input:

A positive Integer n

Output:

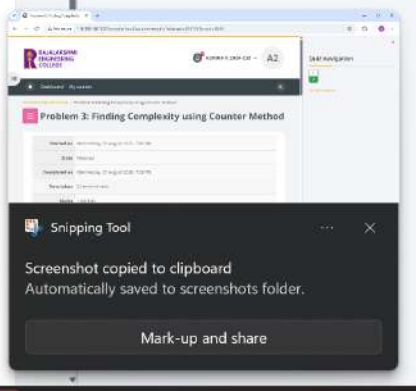
Print the value of the counter variable

Answer:

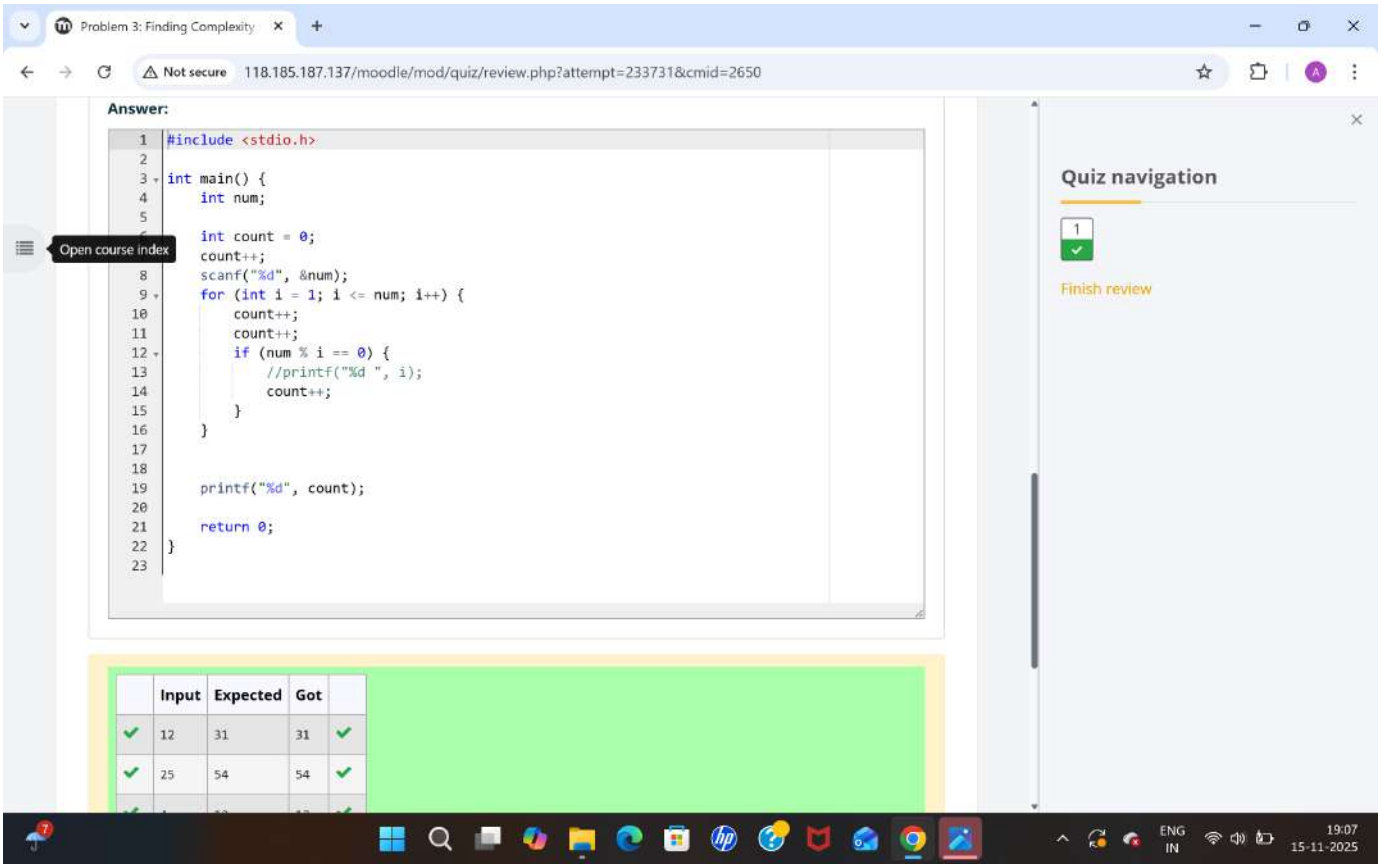
Quiz navigation

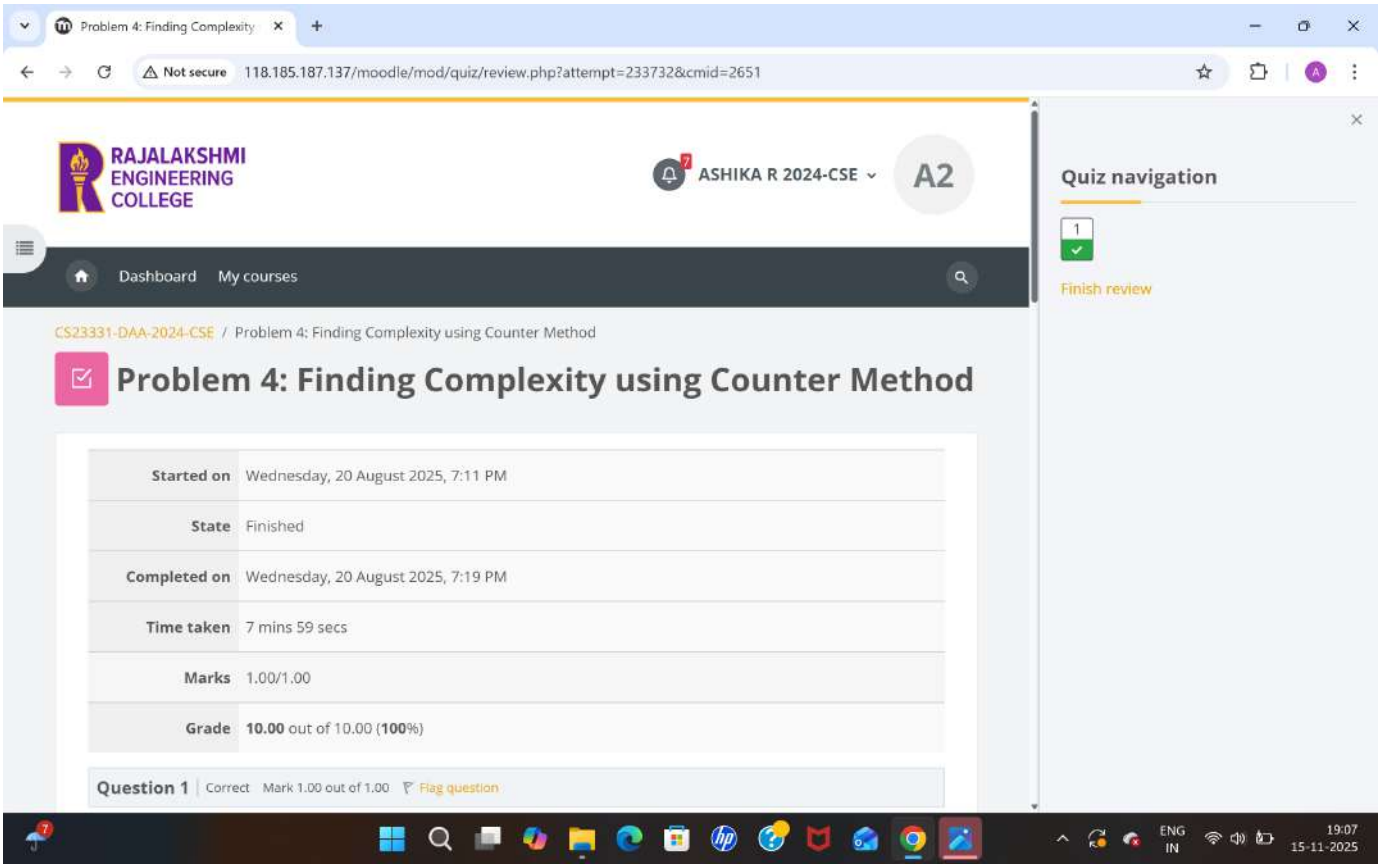
1

Finish review



ENG IN 19:06 15-11-2025


Problem 3: Finding Complexity
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233731&cmid=2650
Answer:
#include <stdio.h>
int main() {
int num;
int count = 0;
count++;
scanf("%d", &num);
for (int i = 1; i <= num; i++) {
count++;
count++;
if (num % i == 0) {
//printf("%d ", i);
count++;
}
}
printf("%d", count);
return 0;
}
Open course index
Quiz navigation
1
Finish review
Input
Expected
Got
12
31
31
25
54
54
ENG
IN
19:07
15-11-2025


Problem 4: Finding Complexity
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233732&cmid=2651
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / Problem 4: Finding Complexity using Counter Method
Problem 4: Finding Complexity using Counter Method
Started on Wednesday, 20 August 2025, 7:11 PM
State Finished
Completed on Wednesday, 20 August 2025, 7:19 PM
Time taken 7 mins 59 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
ENG IN
19:07
15-11-2025

Problem 4: Finding Complexity

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=233732&cmid=2651

Grade **10.00** out of 10.00 (**100**%)

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

Open course index

Convert the following algorithm into a program and find its time complexity using counter method.

```
void function(int n)
{
    int c= 0;
    for(int i=n/2; i<n; i++)
        for(int j=1; j<n; j = 2 * j)
            for(int k=1; k<n; k = k * 2)
                c++;
}
```

Note: No need of counter increment for declarations and scanf() and count variable printf() statements.

Input:

A positive Integer n

Output:

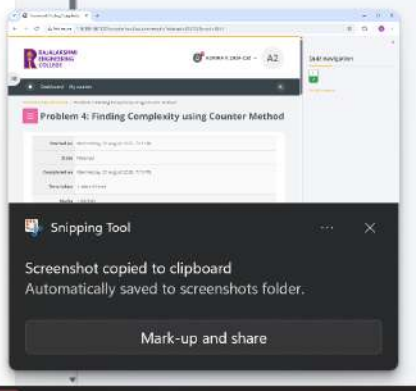
Print the value of the counter variable

Answer:

Quiz navigation

1

Finish review



ENG IN 19:07 15-11-2025

Problem 4: Finding Complexity

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=233732&cmid=2651

Answer:

```c
#include <stdio.h>

int main() {
    int n;
    scanf("%d",&n);

    int c = 0;
    c++;

    for (int i = n / 2; i < n; i++) {
        c++;
        for (int j = 1; j < n; j = 2 * j) {
            c++;
            for (int k = 1; k < n; k = k * 2) {
                c++;
                c++;
            }
            c++;
        }
        c++;
    }
    c++;


    printf("%d\n", c);

    return 0;
}
```

Open course index

Quiz navigation

1

Finish review

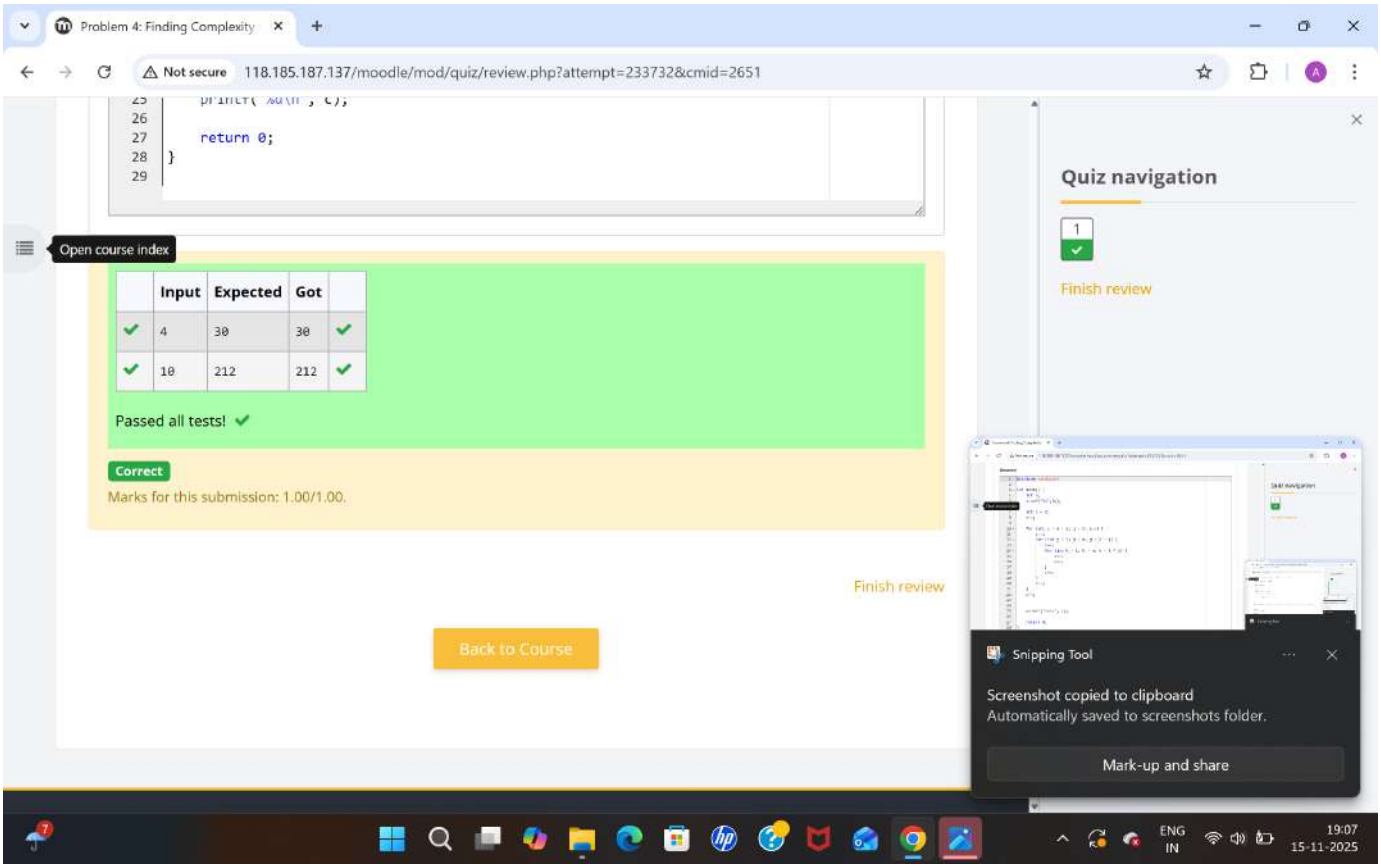
Input Expected Got

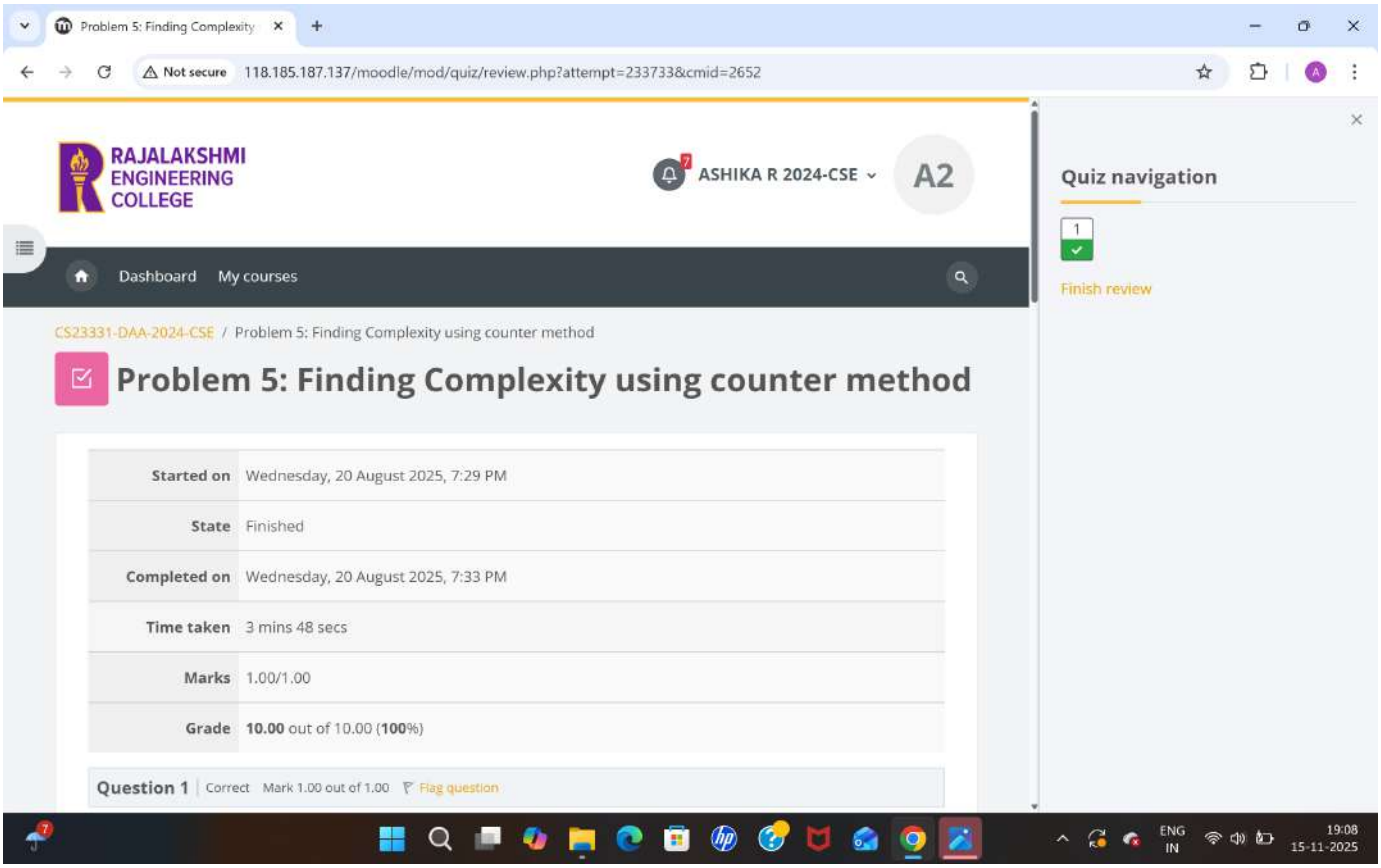
Snipping Tool

Screenshot copied to clipboard
Automatically saved to screenshots folder.

Mark-up and share

ENG IN 19:07 15-11-2025


Problem 4: Finding Complexity
Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=233732&cmid=2651
return 0;
}
Open course index
Input
Expected
Got
4
30
30
10
212
212
Passed all tests!
Correct
Marks for this submission: 1.00/1.00.
Finish review
Back to Course
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:07
15-11-2025


Problem 5: Finding Complexity
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233733&cmid=2652
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / Problem 5: Finding Complexity using counter method
Problem 5: Finding Complexity using counter method
Started on Wednesday, 20 August 2025, 7:29 PM
State Finished
Completed on Wednesday, 20 August 2025, 7:33 PM
Time taken 3 mins 48 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
ENG IN
19:08
15-11-2025

Problem 5: Finding Complexity

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=233733&cmid=2652

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

Open course index

Convert the following algorithm into a program and find its time complexity using counter method.

```
void reverse(int n)
   int rev = 0, remainder;
   while (n != 0)
    {
        remainder = n % 10;
        rev = rev * 10 + remainder;
        n/= 10;

    }
print(rev);
}
```

Note: No need of counter increment for declarations and scanf() and count variable printf() statements.

Input:

A positive Integer n

Output:

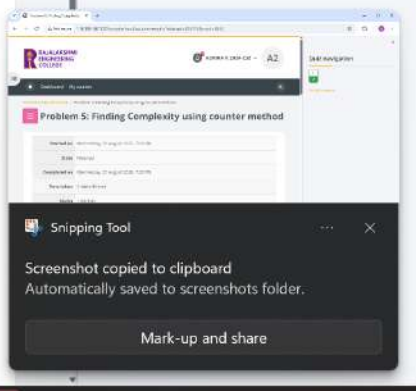
Print the value of the counter variable

Answer:

Quiz navigation

1

Finish review



ENG IN 19:08 15-11-2025

Problem 5: Finding Complexity

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=233733&cmid=2652

Answer:

```c
#include <stdio.h>

int main() {
    int n;
    scanf("%d", &n);

    int rev = 0, remainder;
    int count = 0;
    count++;

    while (n != 0) {
        remainder = n % 10;
        count++;
        rev = rev * 10 + remainder;
        count++;
        n /= 10;
        count++;
        count++;
    }

    count++;
    count++;
    printf("%d\n", count);

    return 0;
}
```

Open course index

Quiz navigation

1

Finish review

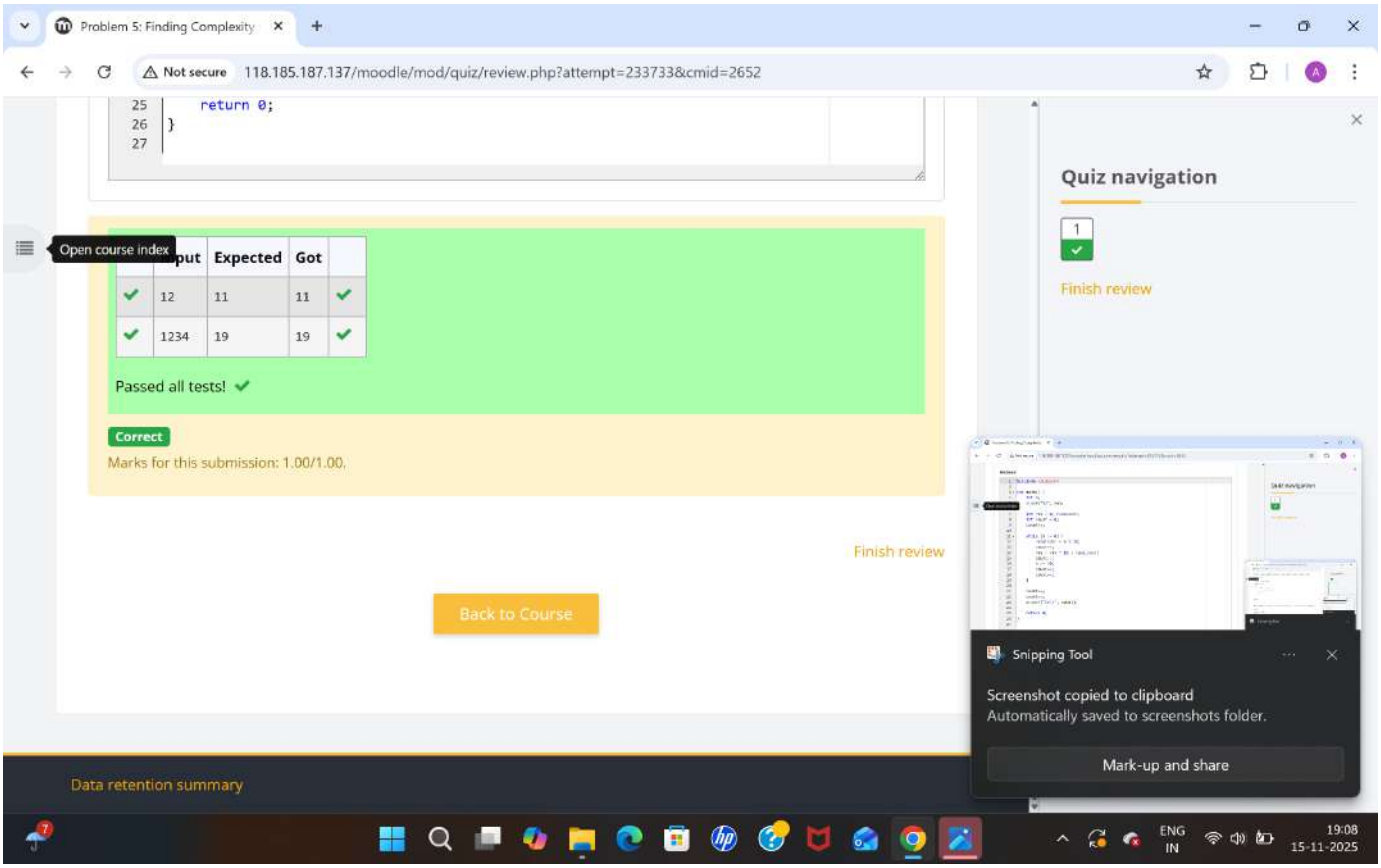
	Input	Expected	Got	

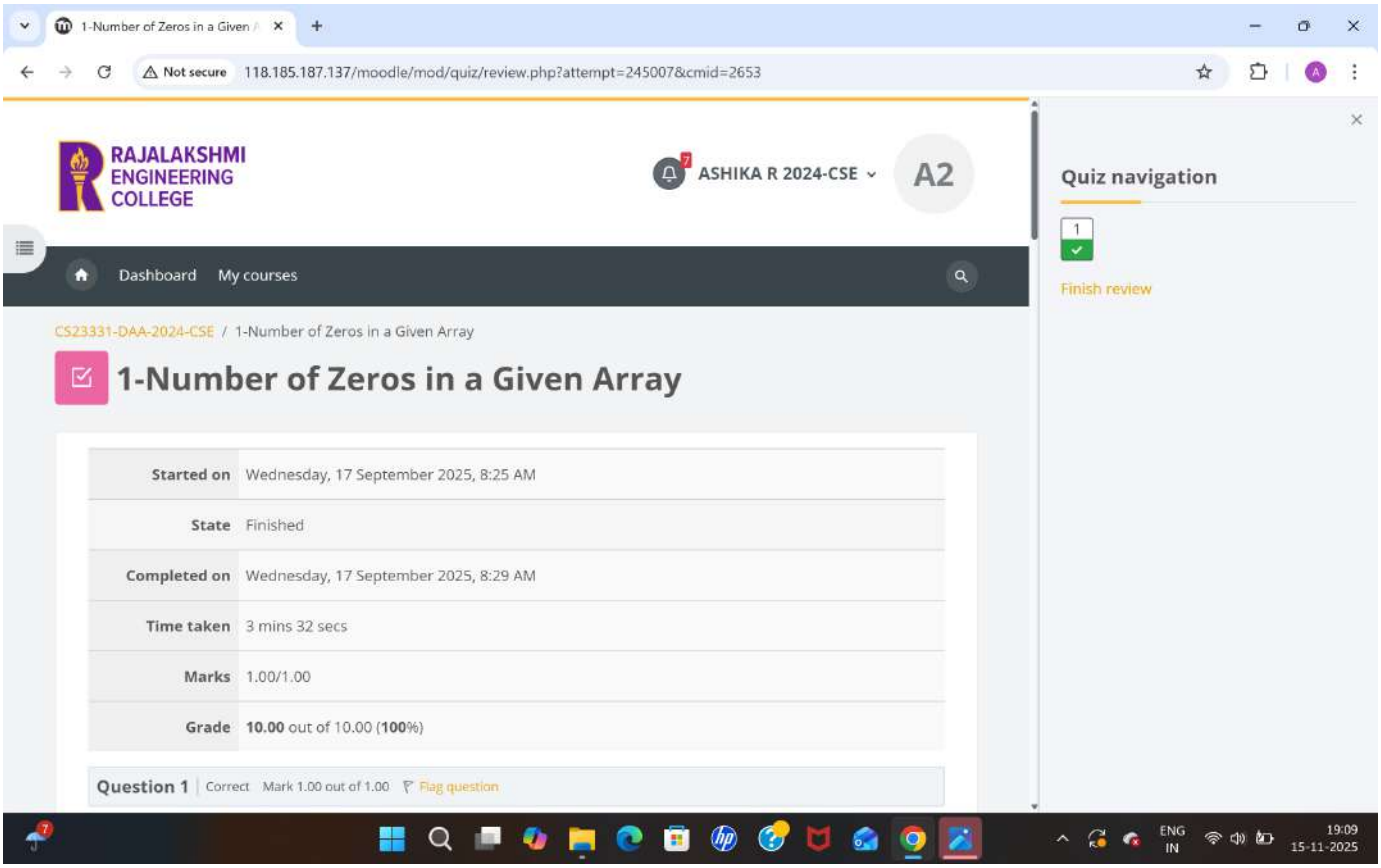
Snipping Tool

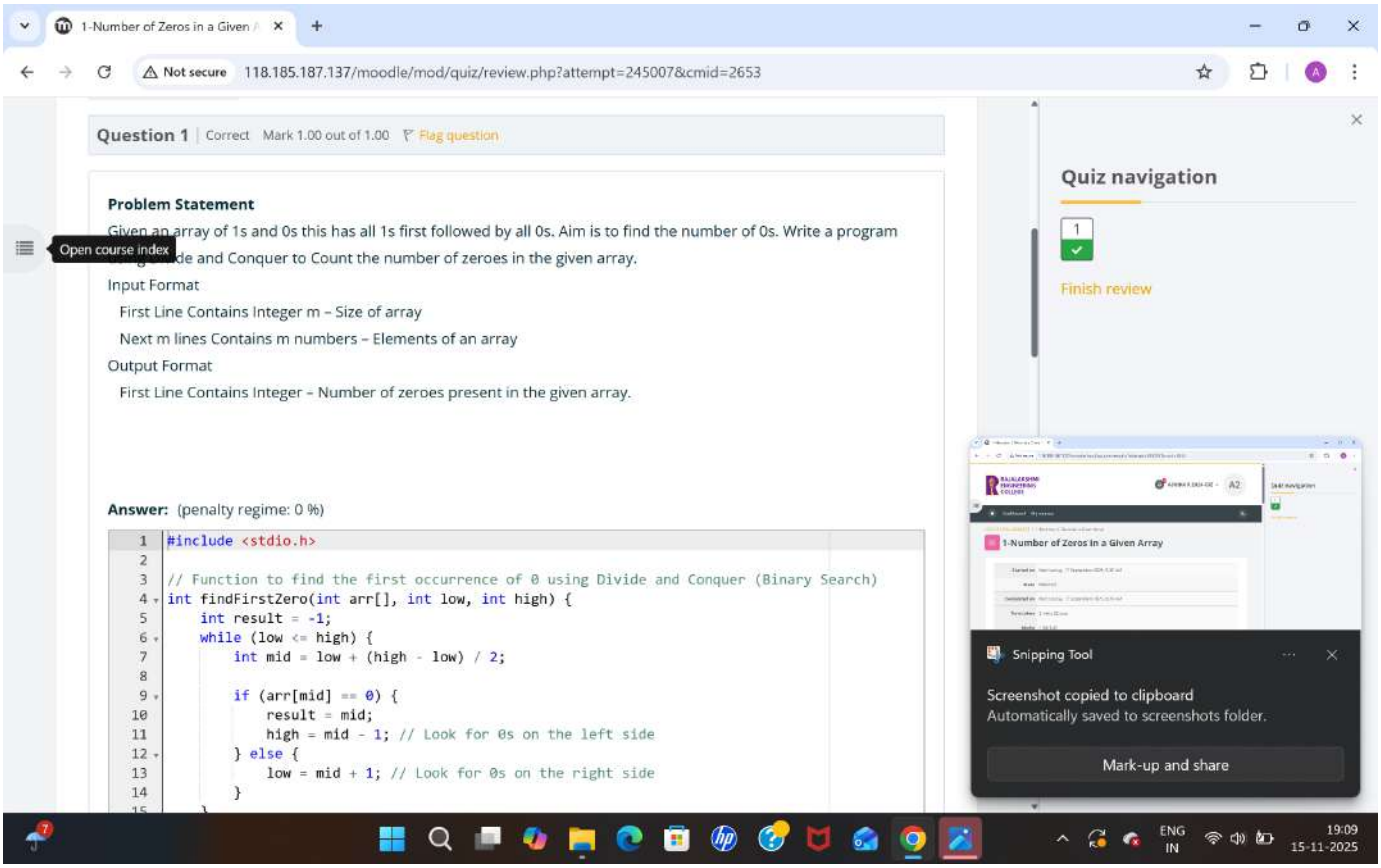
Screenshot copied to clipboard
Automatically saved to screenshots folder.

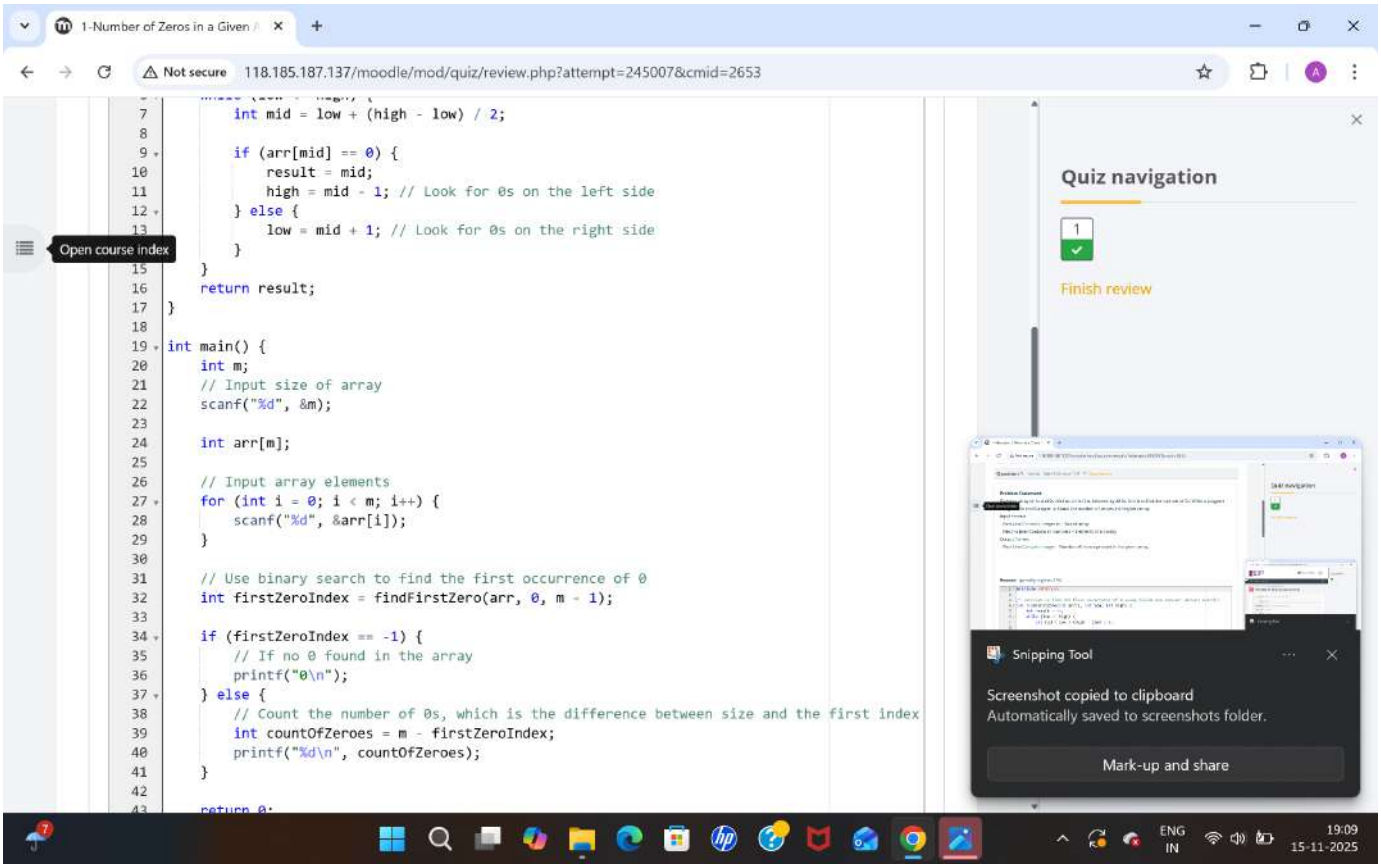
Mark-up and share

ENG IN 19:08 15-11-2025

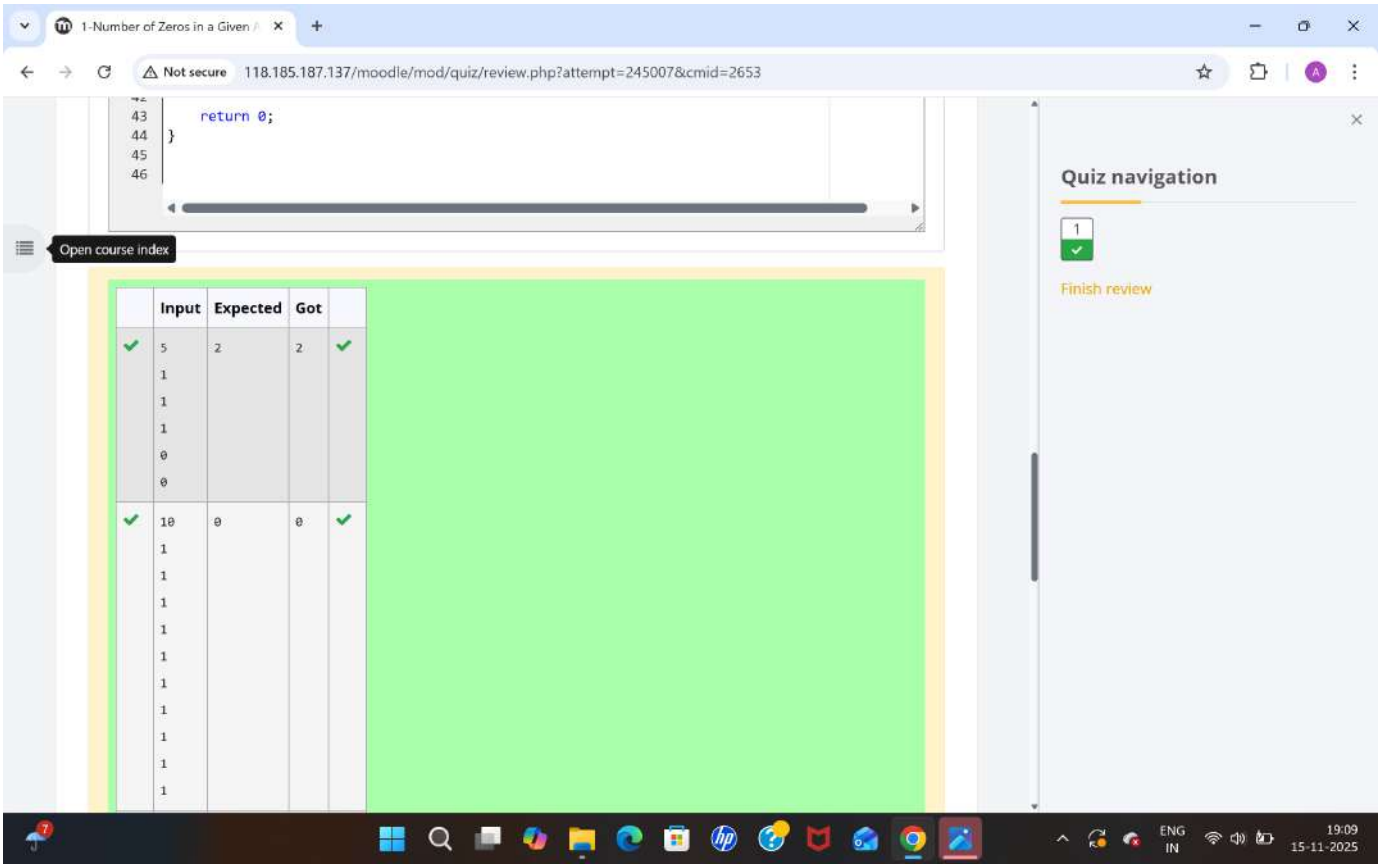

Problem 5: Finding Complexity
Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=233733&cmid=2652
return 0;
}
Open course index
Expected
Got
12
11
11
1234
19
19
Passed all tests!
Correct
Marks for this submission: 1.00/1.00.
Finish review
Back to Course
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Data retention summary
ENG
IN
19:08
15-11-2025

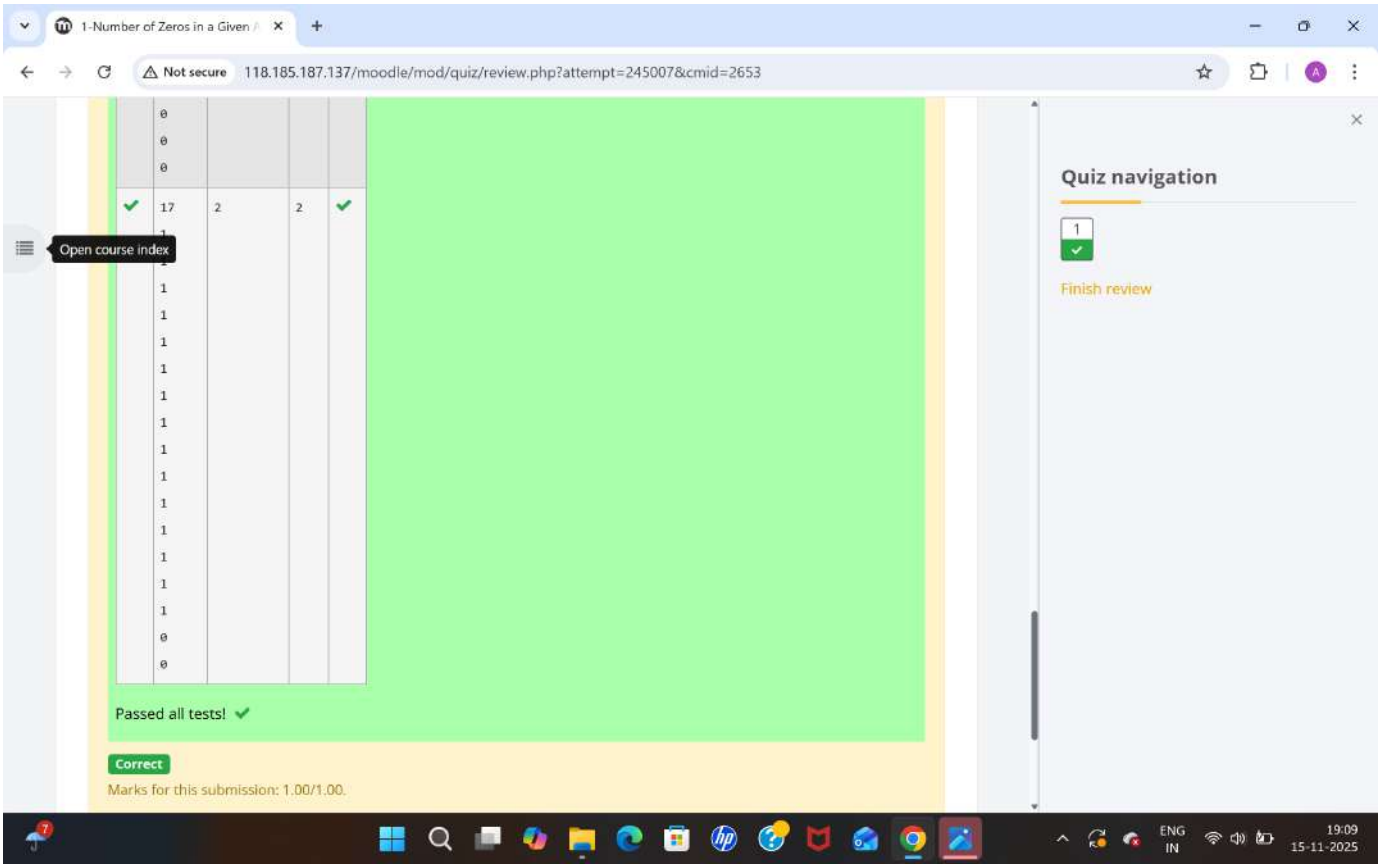

1-Number of Zeros in a Given
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245007&cmid=2653
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 1-Number of Zeros in a Given Array
1-Number of Zeros in a Given Array
Started on Wednesday, 17 September 2025, 8:25 AM
State Finished
Completed on Wednesday, 17 September 2025, 8:29 AM
Time taken 3 mins 32 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
ENG IN
19:09
15-11-2025

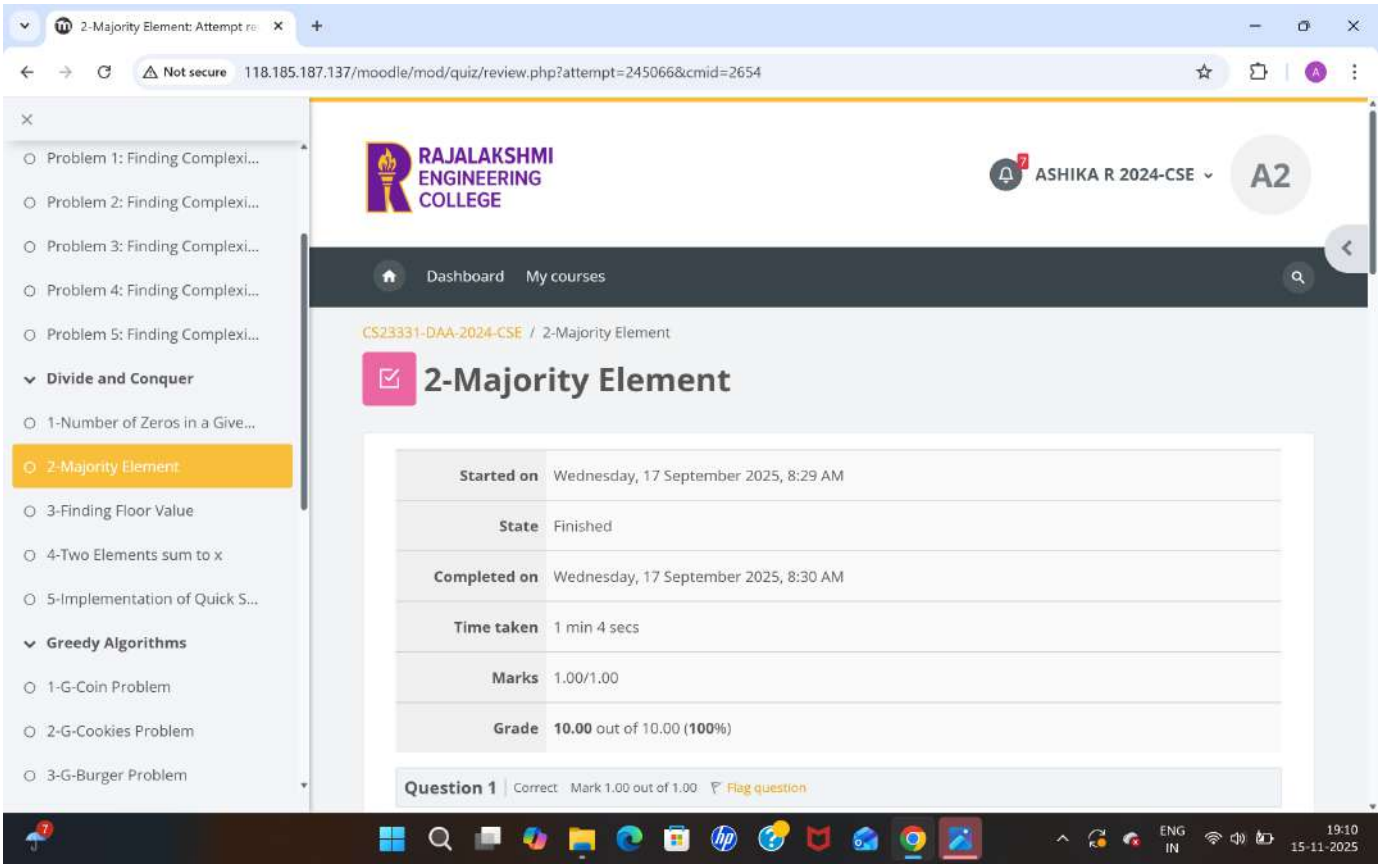

1-Number of Zeros in a Given
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245007&cmid=2653
Question 1 | Correct Mark 1.00 out of 1.00 Flag question
Problem Statement
Given an array of 1s and 0s this has all 1s first followed by all 0s. Aim is to find the number of 0s. Write a program
de and Conquer to Count the number of zeroes in the given array.
Input Format
First Line Contains Integer m – Size of array
Next m lines Contains m numbers – Elements of an array
Output Format
First Line Contains Integer – Number of zeroes present in the given array.
Open course index
Answer: (penalty regime: 0 %)
#include <stdio.h>
// Function to find the first occurrence of 0 using Divide and Conquer (Binary Search)
int findFirstZero(int arr[], int low, int high) {
int result = -1;
while (low <= high) {
int mid = low + (high - low) / 2;
if (arr[mid] == 0) {
result = mid;
high = mid - 1; // Look for 0s on the left side
} else {
low = mid + 1; // Look for 0s on the right side
}
Quiz navigation
1
Finish review
1-Number of Zeros in a Given Array
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:09
15-11-2025

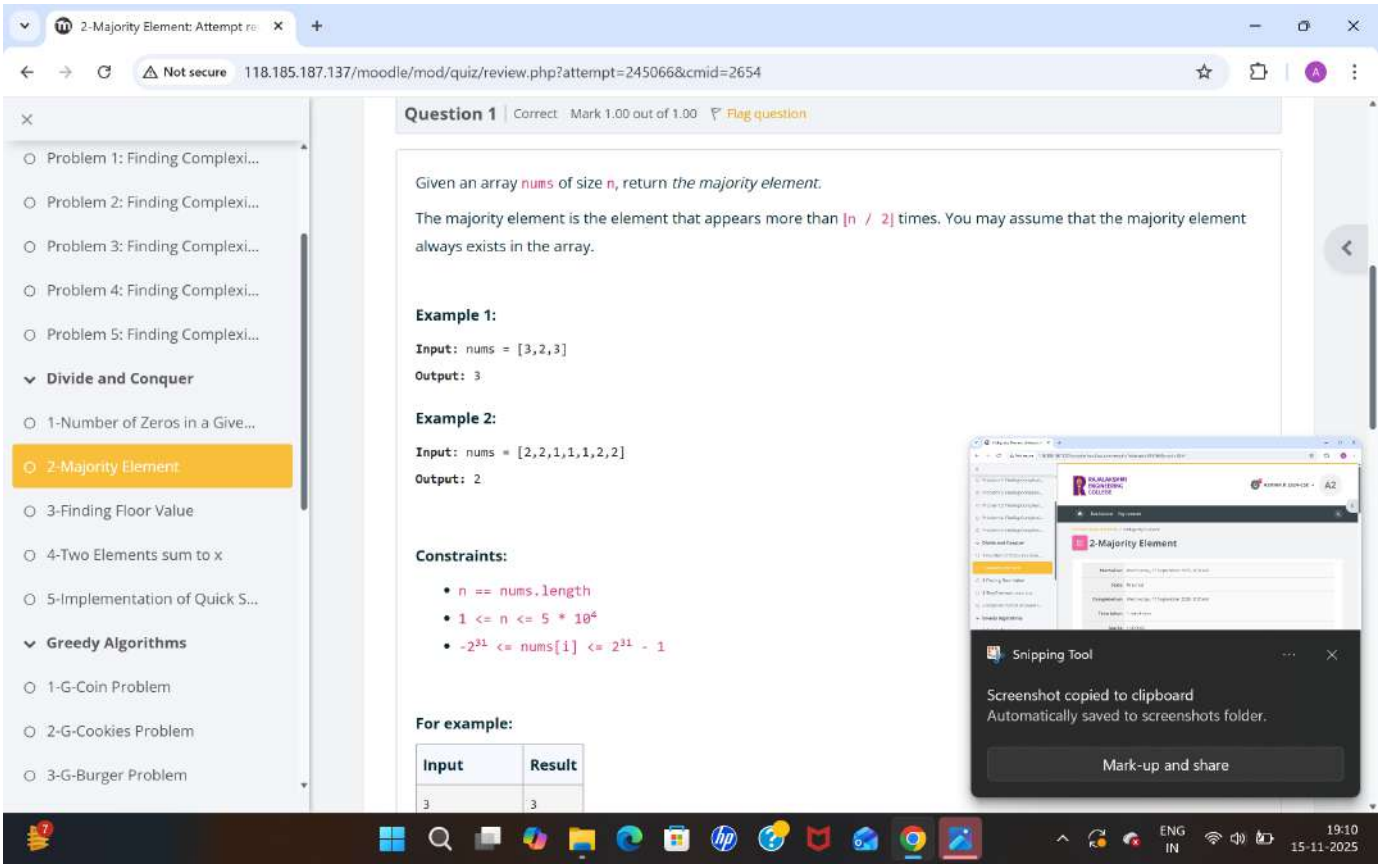


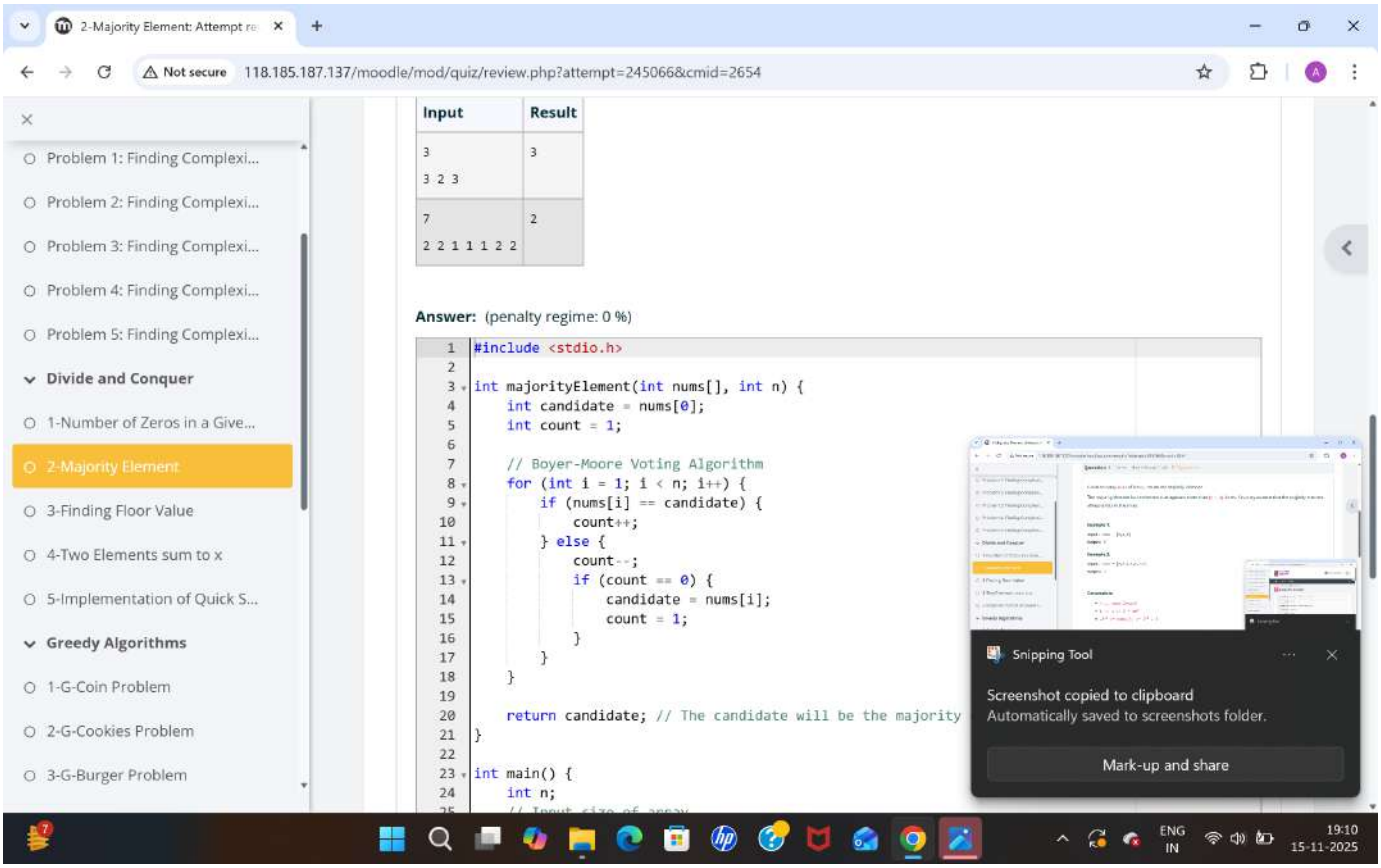

```
        int mid = low + (high - low) / 2;

        if (arr[mid] == 0) {
            result = mid;
            high = mid - 1; // Look for 0s on the left side
        } else {
            low = mid + 1; // Look for 0s on the right side
        }
    }
    return result;
}

int main() {
    int m;
    // Input size of array
    scanf("%d", &m);

    int arr[m];

    // Input array elements
    for (int i = 0; i < m; i++) {
        scanf("%d", &arr[i]);
    }

    // Use binary search to find the first occurrence of 0
    int firstZeroIndex = findFirstZero(arr, 0, m - 1);

    if (firstZeroIndex == -1) {
        // If no 0 found in the array
        printf("0\n");
    } else {
        // Count the number of 0s, which is the difference between size and the first index
        int countOfZeroes = m - firstZeroIndex;
        printf("%d\n", countOfZeroes);
    }

    return 0;
```

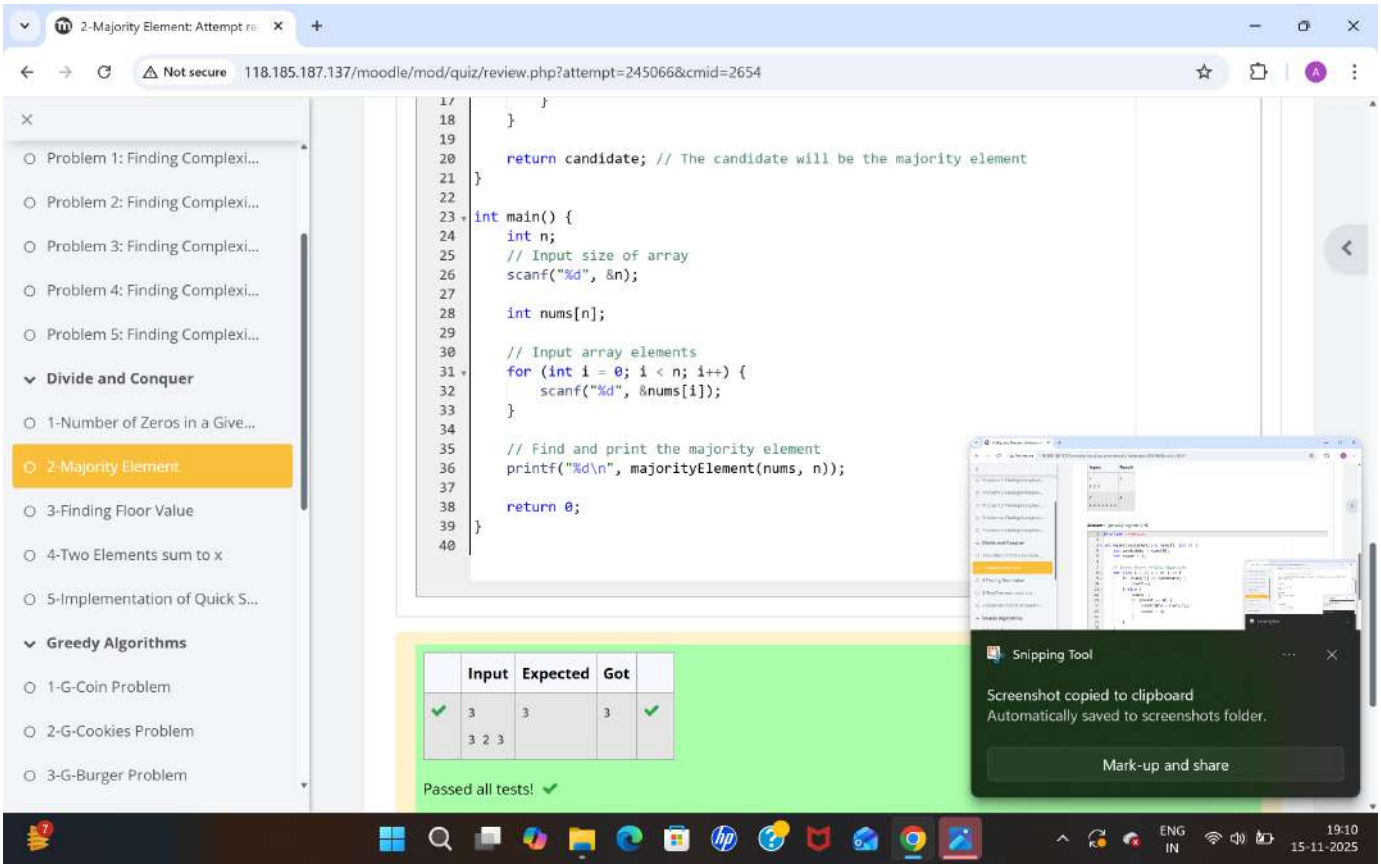

1-Number of Zeros in a Given
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245007&cmid=2653
return 0;
}
Open course index
Input
Expected
Got
5 1 1 1 0 0
2
2
10 1 1 1 1 1 1 1 1 1
0
0
Quiz navigation
1
Finish review
ENG IN
19:09
15-11-2025

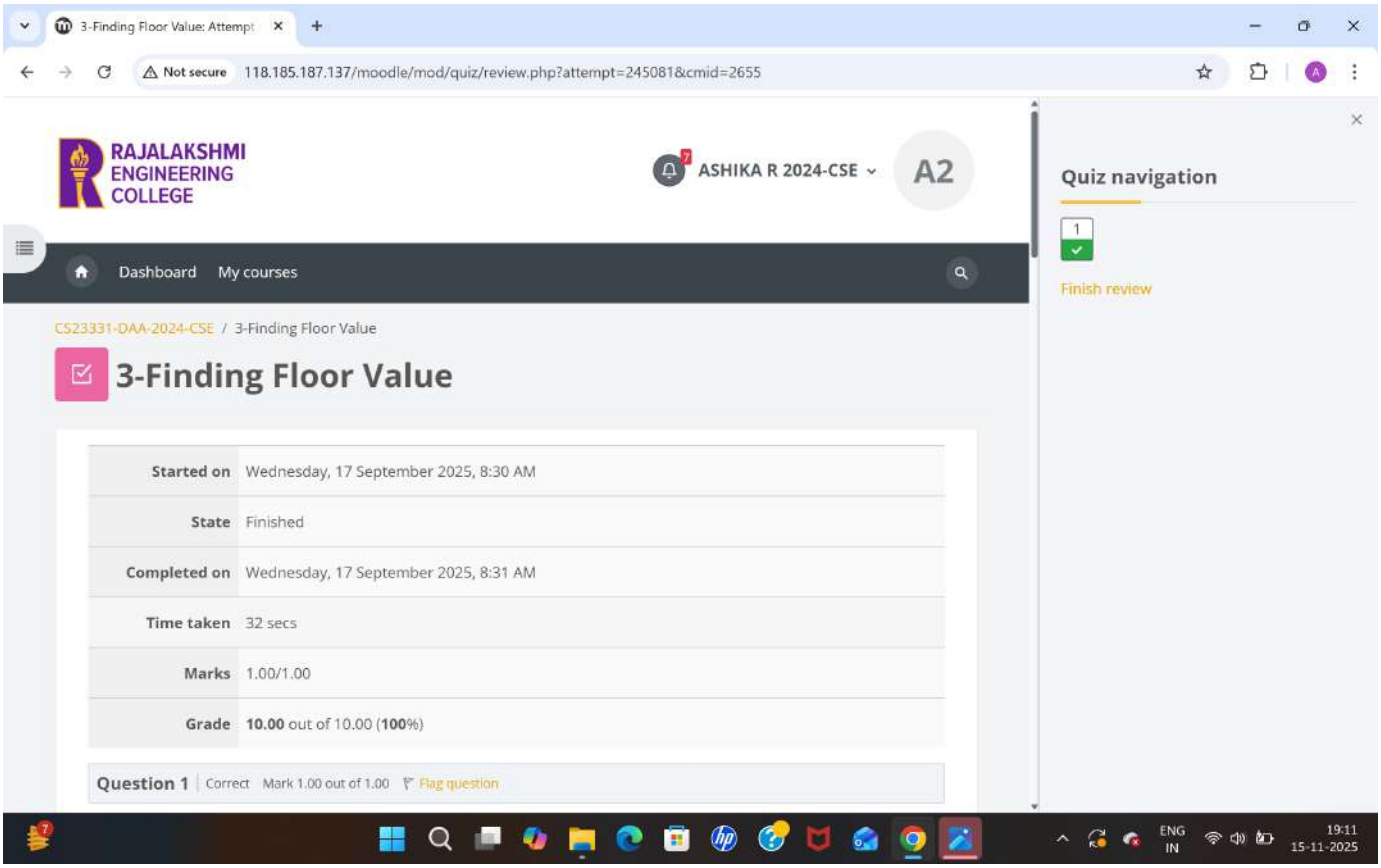

1-Number of Zeros in a Given
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245007&cmid=2653
Open course index
Quiz navigation
1
Finish review
Passed all tests!
Correct
Marks for this submission: 1.00/1.00.
ENG
IN
19:09
15-11-2025


2-Majority Element: Attempt re
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245066&cmid=2654
Problem 1: Finding Complexi...
Problem 2: Finding Complexi...
Problem 3: Finding Complexi...
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
Greedy Algorithms
1-G-Coin Problem
2-G-Cookies Problem
3-G-Burger Problem
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 2-Majority Element
2-Majority Element
Started on
Wednesday, 17 September 2025, 8:29 AM
State
Finished
Completed on
Wednesday, 17 September 2025, 8:30 AM
Time taken
1 min 4 secs
Marks
1.00/1.00
Grade
10.00 out of 10.00 (100%)
Question 1
Correct
Mark 1.00 out of 1.00
Flag question
ENG
IN
19:10
15-11-2025


2-Majority Element: Attempt re
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245066&cmid=2654
Problem 1: Finding Complexi...
Problem 2: Finding Complexi...
Problem 3: Finding Complexi...
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
Greedy Algorithms
1-G-Coin Problem
2-G-Cookies Problem
3-G-Burger Problem
Question 1
Correct
Mark 1.00 out of 1.00
Flag question
Given an array nums of size n, return the majority element.
The majority element is the element that appears more than ⌊n / 2⌋ times. You may assume that the majority element always exists in the array.
Example 1:
Input: nums = [3,2,3]
Output: 3
Example 2:
Input: nums = [2,2,1,1,1,2,2]
Output: 2
Constraints:
n == nums.length
1 <= n <= 5 * 10^4
-2^31 <= nums[i] <= 2^31 - 1
For example:
Input
Result
3
3
2-Majority Element
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:10
15-11-2025


2-Majority Element: Attempt re
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245066&cmid=2654
Problem 1: Finding Complexi...
Problem 2: Finding Complexi...
Problem 3: Finding Complexi...
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
Greedy Algorithms
1-G-Coin Problem
2-G-Cookies Problem
3-G-Burger Problem
Input
Result
3
3 2 3
3
7
2 2 1 1 1 2 2
2
Answer: (penalty regime: 0 %)
#include <stdio.h>
int majorityElement(int nums[], int n) {
int candidate = nums[0];
int count = 1;
// Boyer-Moore Voting Algorithm
for (int i = 1; i < n; i++) {
if (nums[i] == candidate) {
count++;
} else {
count--;
if (count == 0) {
candidate = nums[i];
count = 1;
}
}
}
return candidate; // The candidate will be the majority
}
int main() {
int n;
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:10
15-11-2025


2-Majority Element: Attempt re
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245066&cmid=2654
Problem 1: Finding Complexi...
Problem 2: Finding Complexi...
Problem 3: Finding Complexi...
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
Greedy Algorithms
1-G-Coin Problem
2-G-Cookies Problem
3-G-Burger Problem
return candidate; // The candidate will be the majority element
int main() {
int n;
// Input size of array
scanf("%d", &n);
int nums[n];
// Input array elements
for (int i = 0; i < n; i++) {
scanf("%d", &nums[i]);
// Find and print the majority element
printf("%d\n", majorityElement(nums, n));
return 0;
Input
Expected
Got
3
3 2 3
3
3
Passed all tests!
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:10
15-11-2025


3-Finding Floor Value: Attempt
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245081&cmid=2655
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 3-Finding Floor Value
3-Finding Floor Value
Started on Wednesday, 17 September 2025, 8:30 AM
State Finished
Completed on Wednesday, 17 September 2025, 8:31 AM
Time taken 32 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
ENG IN
19:11
15-11-2025

3-Finding Floor Value: Attempt

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=245081&cmid=2655

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

Problem Statement:

Given a sorted array and a value x, the floor of x is the largest element in array smaller than or equal to x. Write divide and conquer algorithm to find floor of x.

Open course index

Input Format

First Line Contains Integer n – Size of array

Next n lines Contains n numbers – Elements of an array

Last Line Contains Integer x – Value for x

Output Format

First Line Contains Integer – Floor value for x

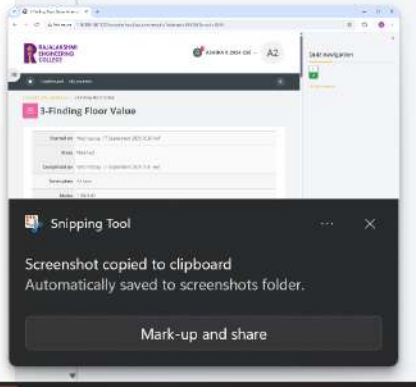
Answer: (penalty regime: 0 %)

```c
#include <stdio.h>

// Function to find the floor of x using binary search
int findFloor(int arr[], int n, int x) {
    int low = 0, high = n - 1;
    int floorValue = -1;  // Initialize the floor as -1 (indicating no floor found)

    while (low <= high) {
        int mid = low + (high - low) / 2;

        // If we find an element equal to x, it's the floor
        if (arr[mid] == x) {
            return arr[mid];
        }
```

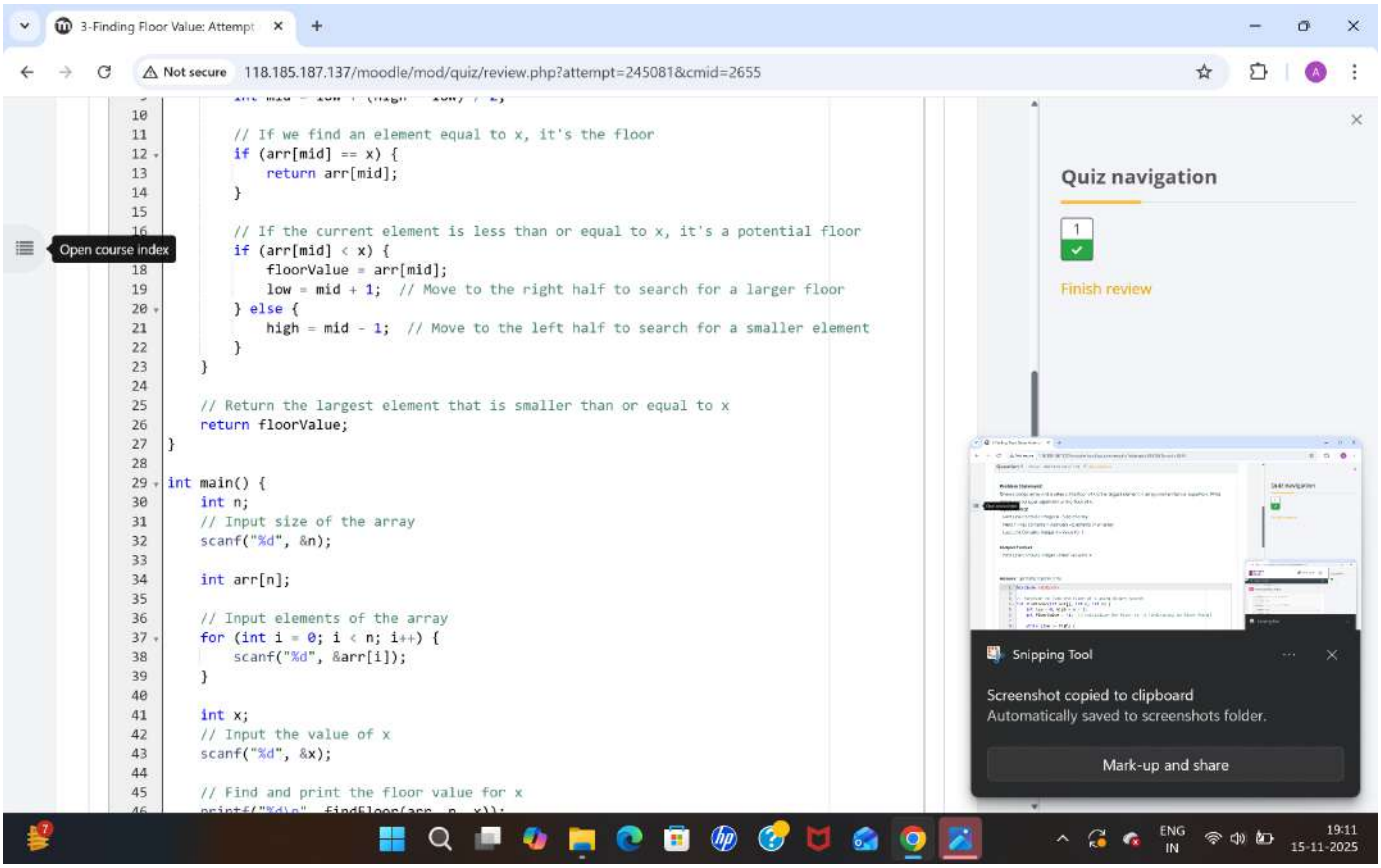
Quiz navigation

1

Finish review



ENG IN 19:11 15-11-2025


3-Finding Floor Value: Attempt
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245081&cmid=2655
Open course index
// If we find an element equal to x, it's the floor
if (arr[mid] == x) {
return arr[mid];
}
// If the current element is less than or equal to x, it's a potential floor
if (arr[mid] < x) {
floorValue = arr[mid];
low = mid + 1; // Move to the right half to search for a larger floor
} else {
high = mid - 1; // Move to the left half to search for a smaller element
}
}
// Return the largest element that is smaller than or equal to x
return floorValue;
}
int main() {
int n;
// Input size of the array
scanf("%d", &n);
int arr[n];
// Input elements of the array
for (int i = 0; i < n; i++) {
scanf("%d", &arr[i]);
}
int x;
// Input the value of x
scanf("%d", &x);
// Find and print the floor value for x
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:11
15-11-2025

3-Finding Floor Value: Attempt

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=245081&cmid=2655

```
41      int x;
42      // Input the value of x
43      scanf("%d", &x);
44
45      // Find and print the floor value for x
46      printf("%d\n", findFloor(arr, n, x));
47
48      return 0;
49  }
50
```

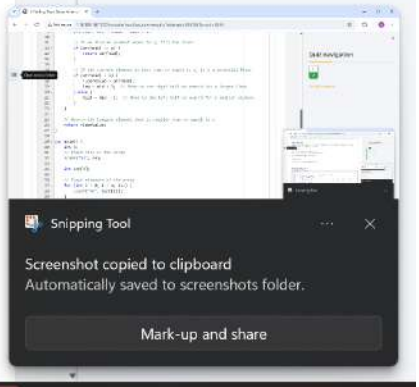
Open course index

	Input	Expected	Got	
✔	6 1 2 8 10 12 19 5	2	2	✔
✔	5 10 22 85 108	85	85	✔

Quiz navigation

1

Finish review

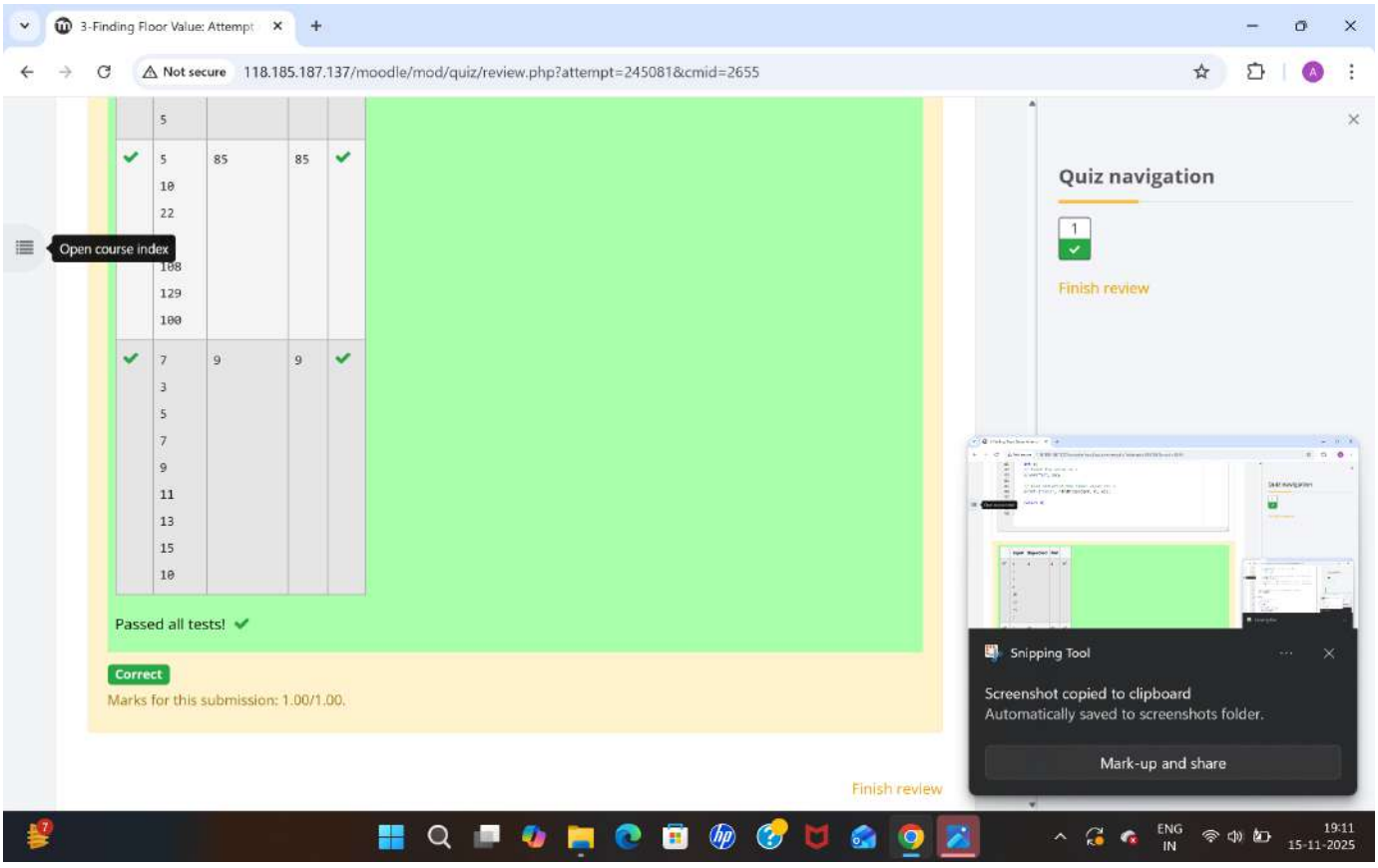


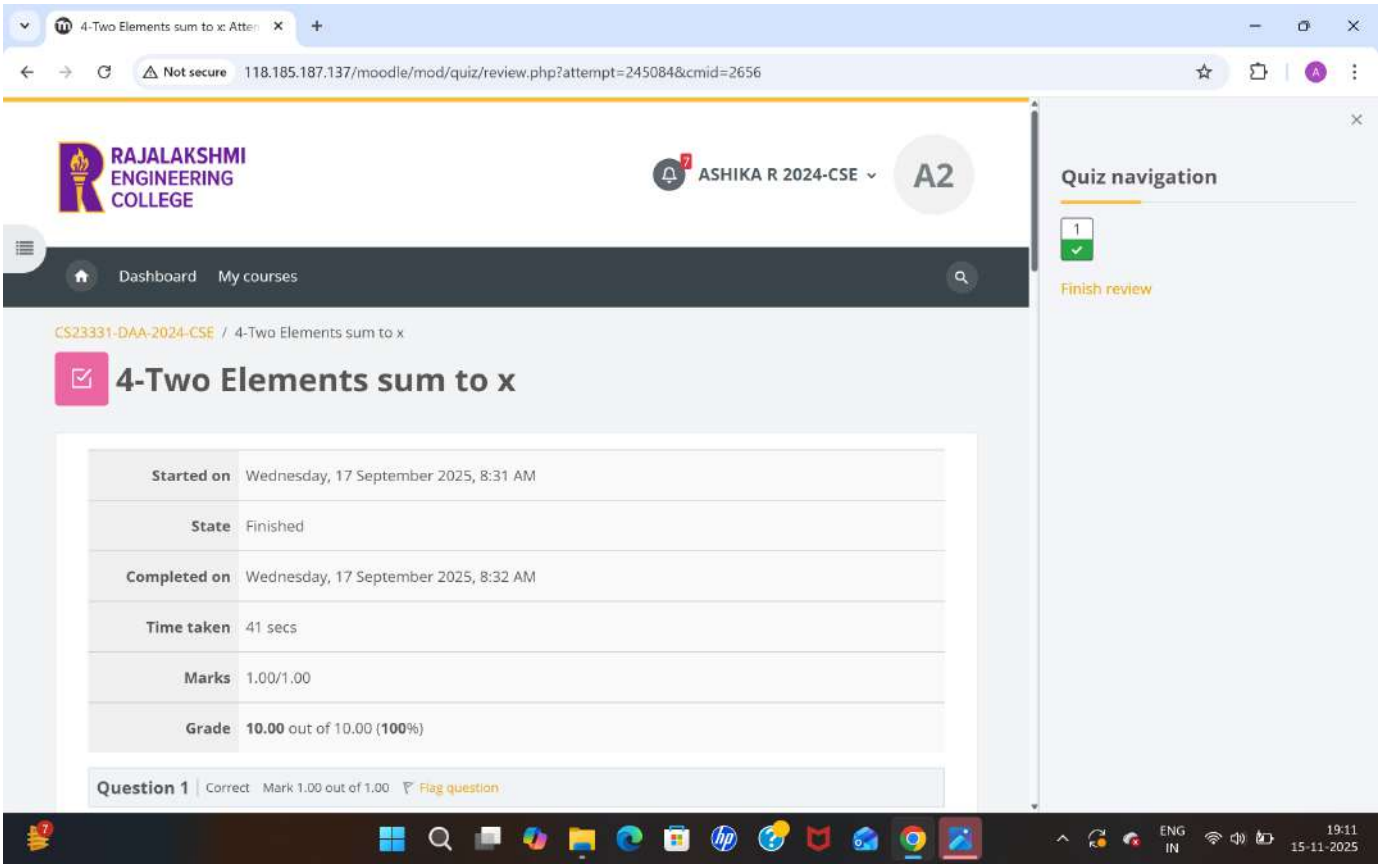
Snipping Tool

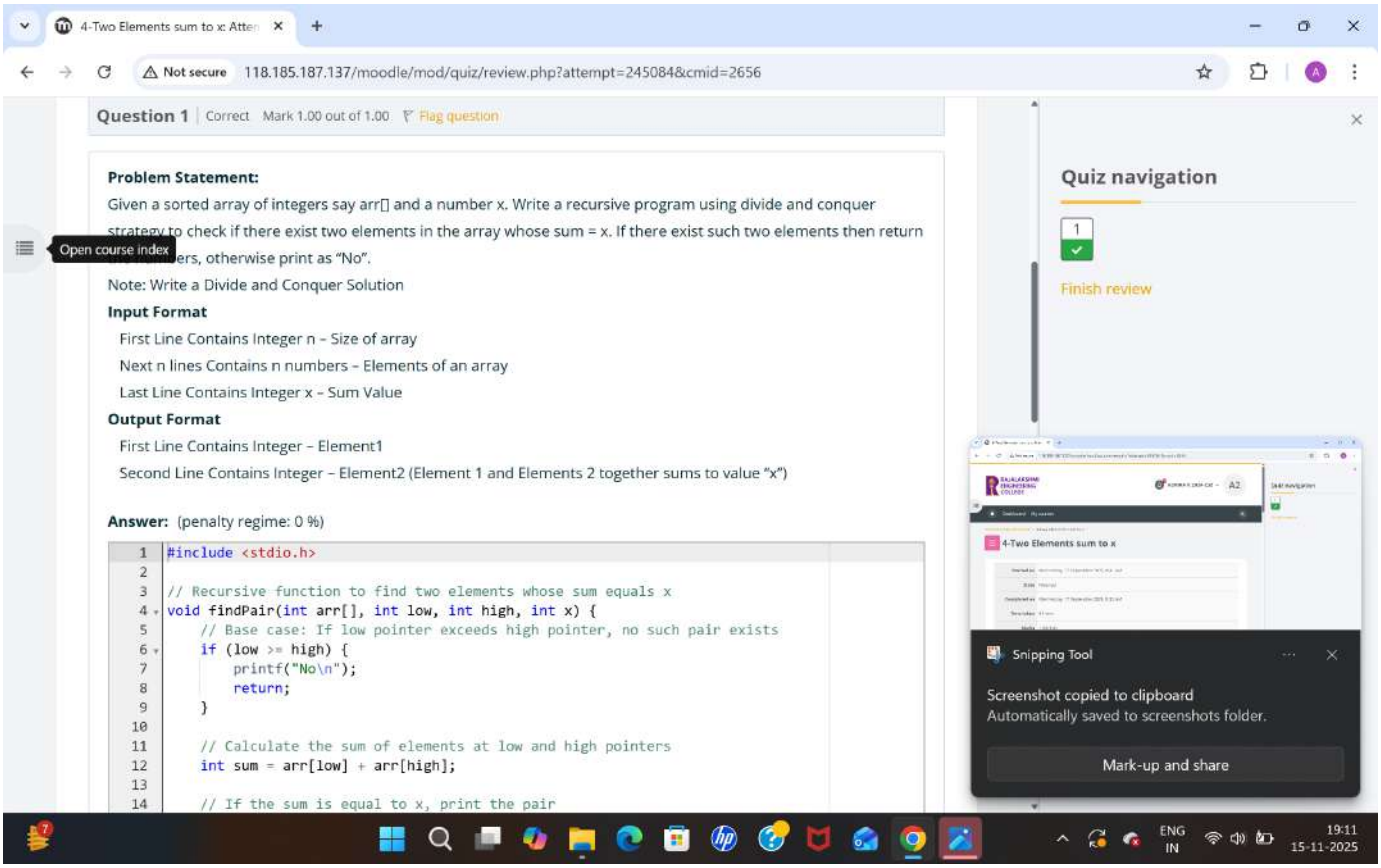
Screenshot copied to clipboard
Automatically saved to screenshots folder.

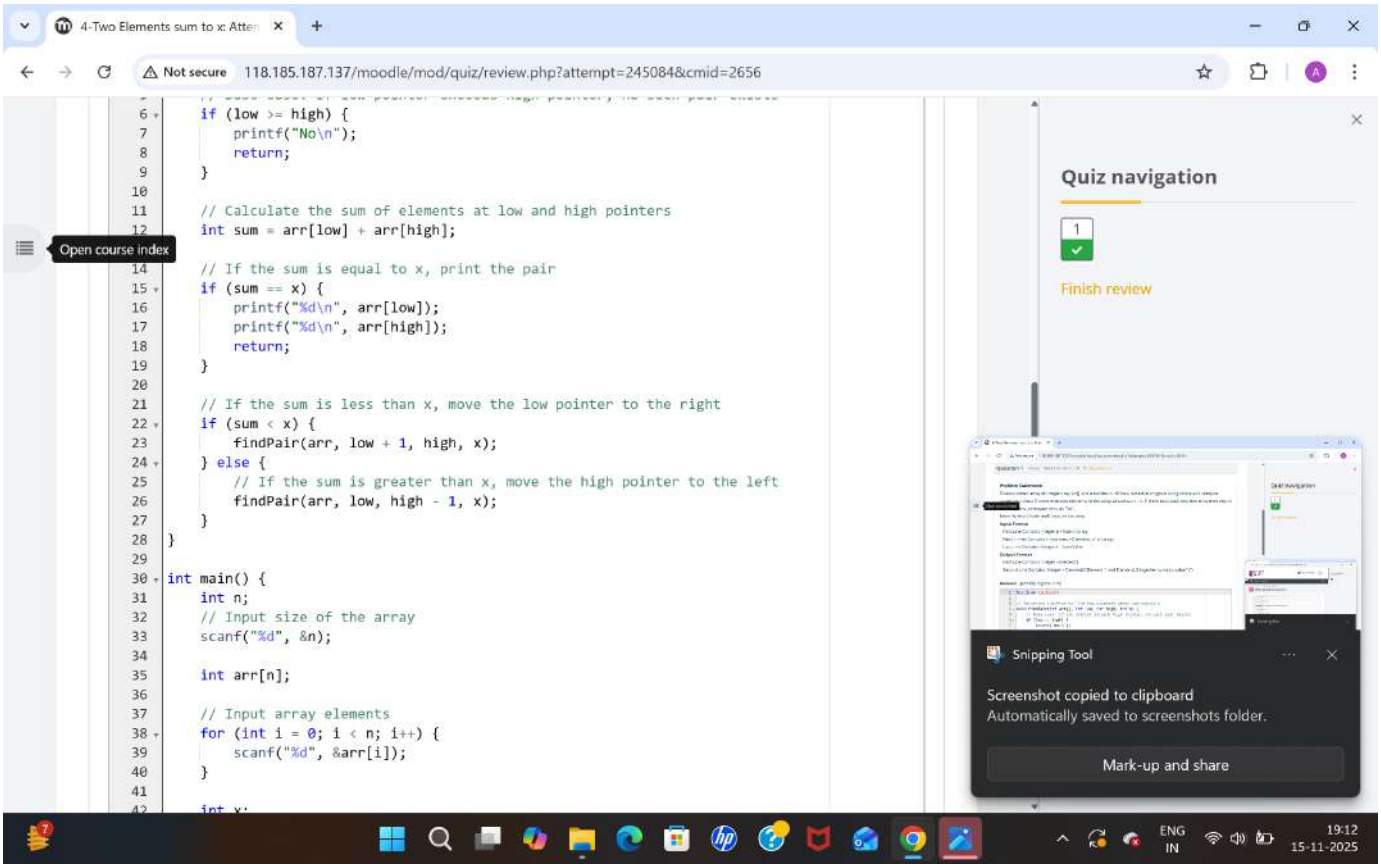
Mark-up and share

ENG IN 19:11 15-11-2025

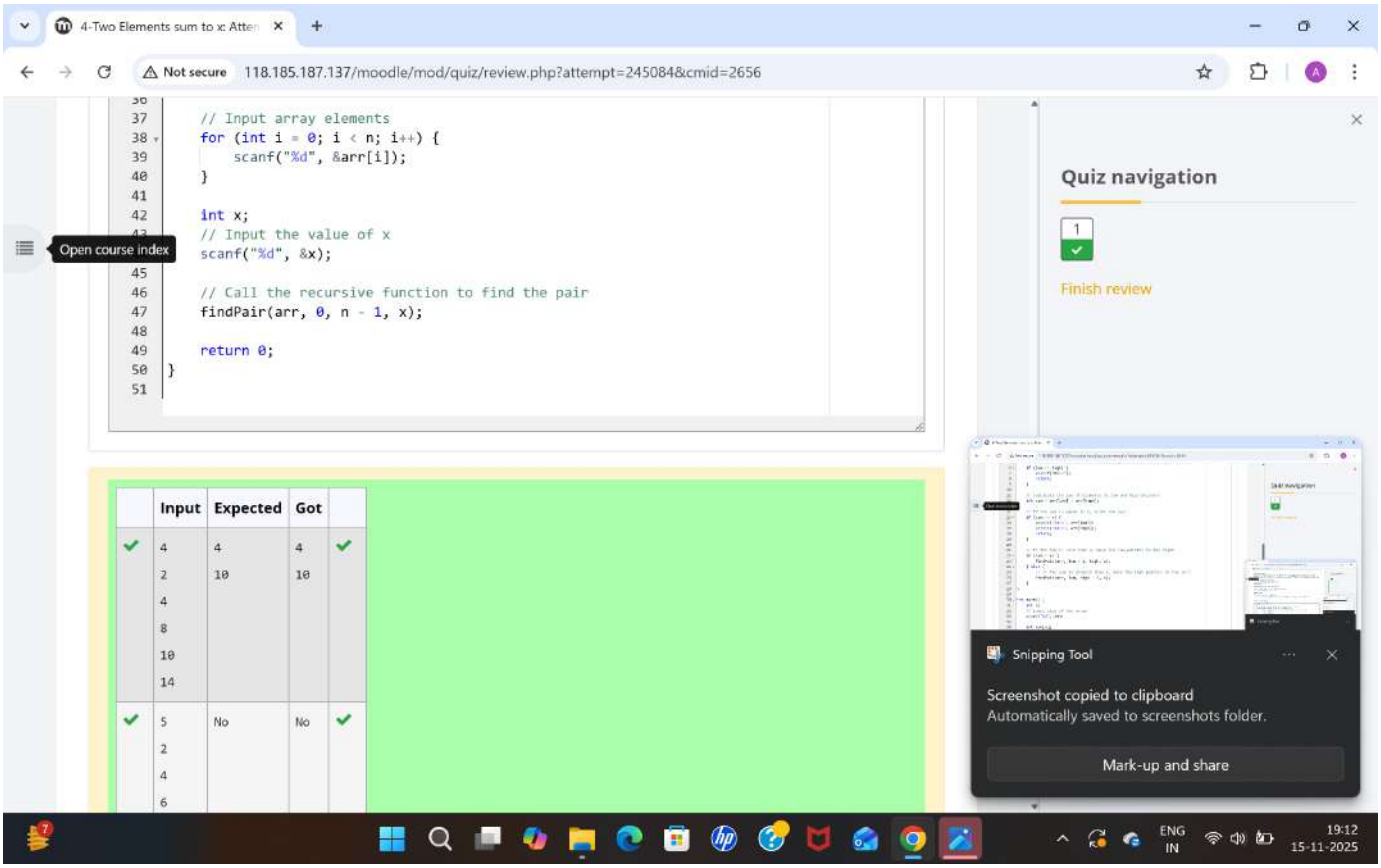

3-Finding Floor Value: Attempt
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245081&cmid=2655
Open course index
5
5 10 22 108 129 100
85
85
7 3 5 7 9 11 13 15 10
9
9
Passed all tests!
Correct
Marks for this submission: 1.00/1.00.
Finish review
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:11
15-11-2025

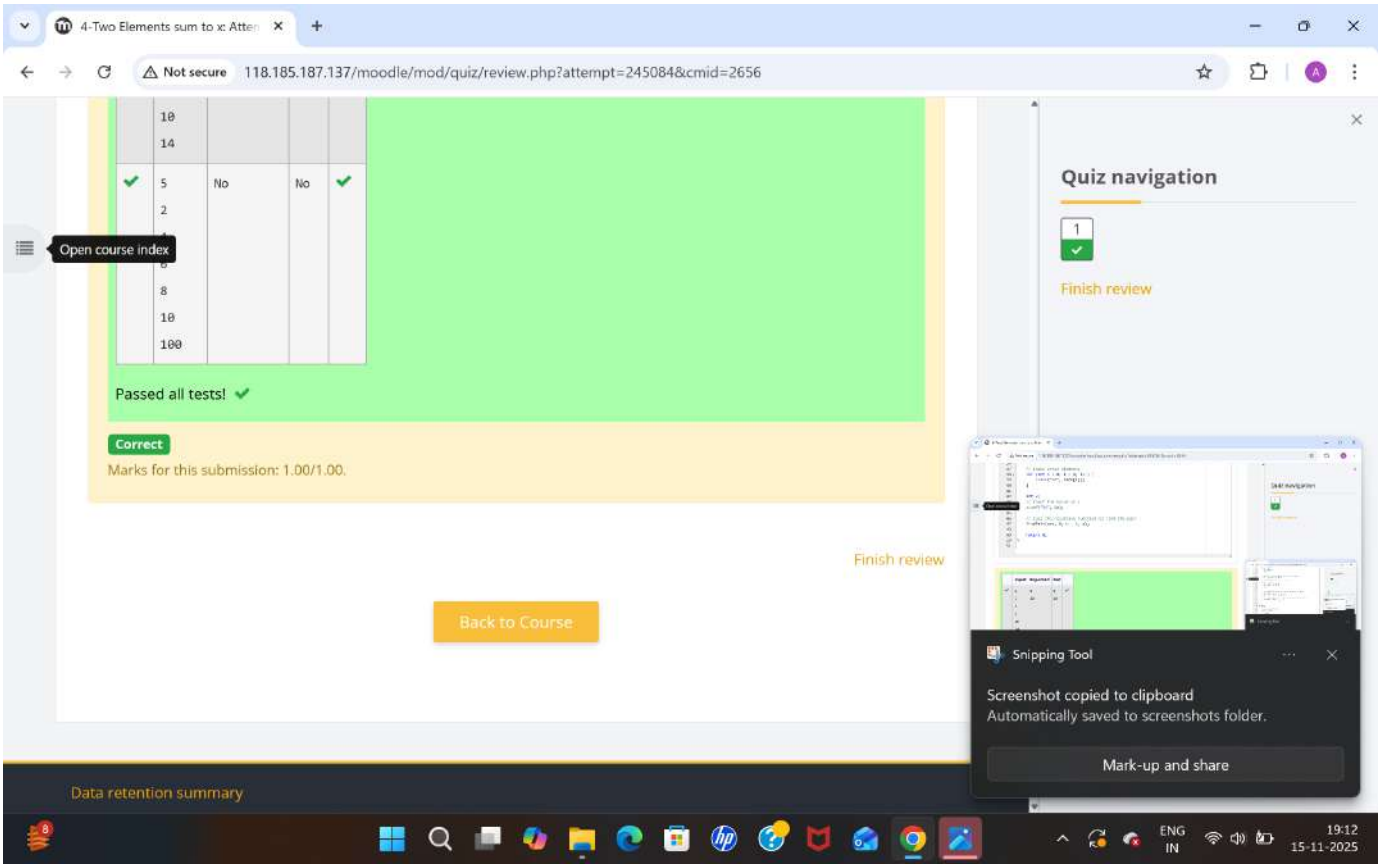

4-Two Elements sum to x: Atter
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245084&cmid=2656
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 4-Two Elements sum to x
4-Two Elements sum to x
Started on Wednesday, 17 September 2025, 8:31 AM
State Finished
Completed on Wednesday, 17 September 2025, 8:32 AM
Time taken 41 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
ENG IN
19:11
15-11-2025

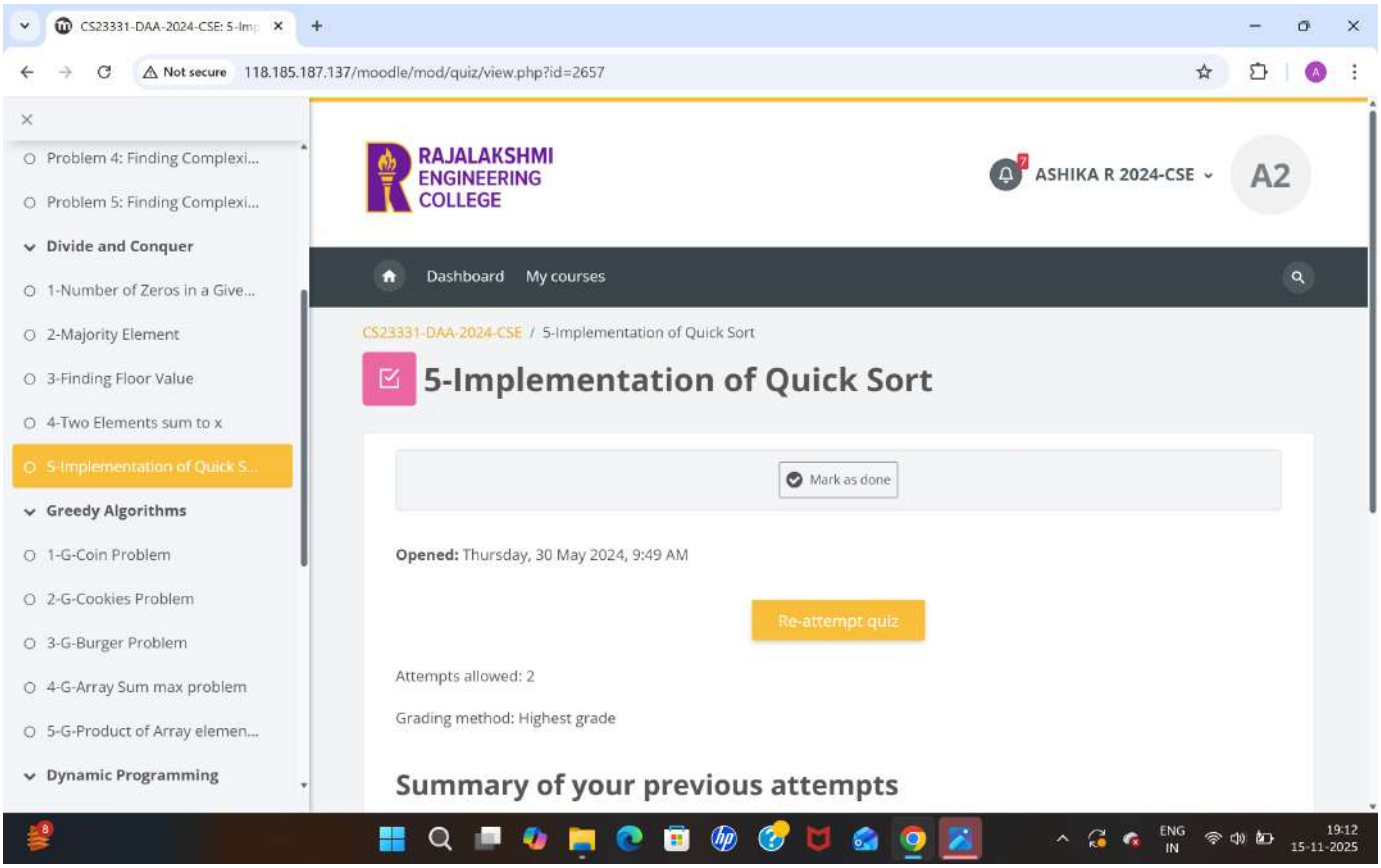

4-Two Elements sum to x: Atten
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245084&cmid=2656
Question 1
Correct
Mark 1.00 out of 1.00
Flag question
Open course index
Problem Statement:
Given a sorted array of integers say arr[] and a number x. Write a recursive program using divide and conquer strategy to check if there exist two elements in the array whose sum = x. If there exist such two elements then return
ers, otherwise print as "No".
Note: Write a Divide and Conquer Solution
Input Format
First Line Contains Integer n – Size of array
Next n lines Contains n numbers – Elements of an array
Last Line Contains Integer x – Sum Value
Output Format
First Line Contains Integer – Element1
Second Line Contains Integer – Element2 (Element 1 and Elements 2 together sums to value "x")
Answer: (penalty regime: 0 %)
#include <stdio.h>
// Recursive function to find two elements whose sum equals x
void findPair(int arr[], int low, int high, int x) {
// Base case: If low pointer exceeds high pointer, no such pair exists
if (low >= high) {
printf("No\n");
return;
}
// Calculate the sum of elements at low and high pointers
int sum = arr[low] + arr[high];
// If the sum is equal to x, print the pair
Quiz navigation
1
Finish review
4-Two Elements sum to x
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:11
15-11-2025

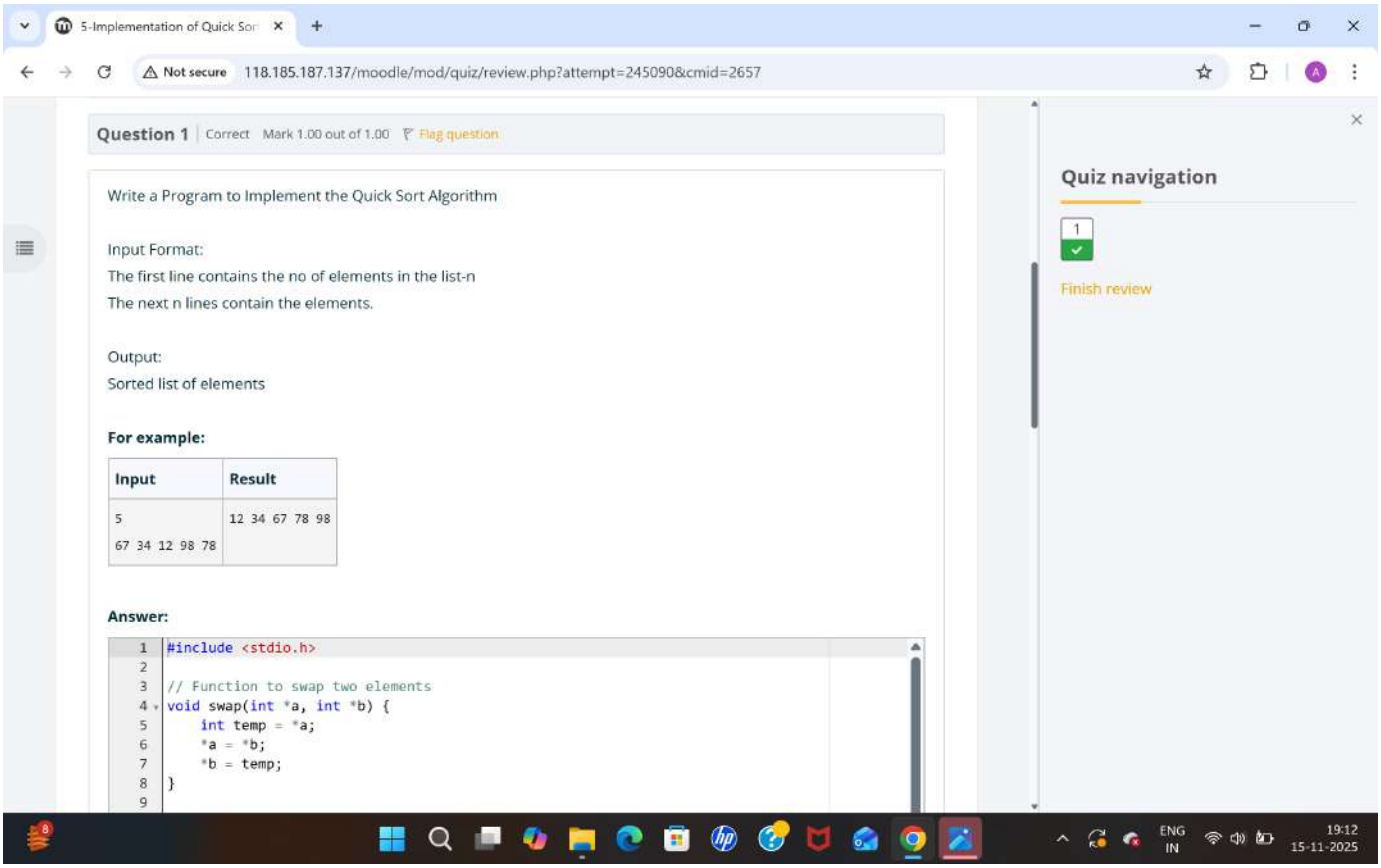


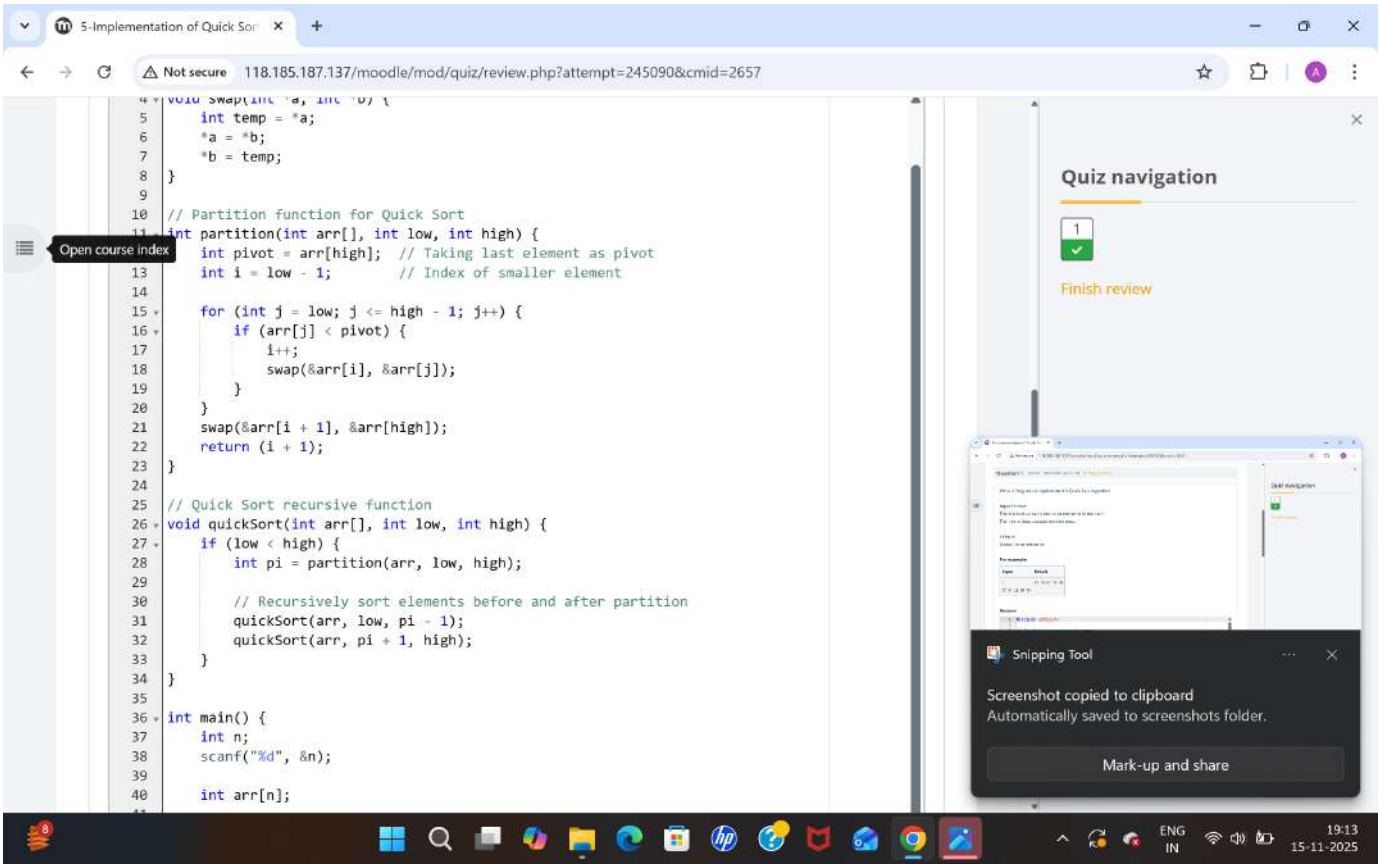

```c
    if (low >= high) {
        printf("No\n");
        return;
    }

    // Calculate the sum of elements at low and high pointers
    int sum = arr[low] + arr[high];

    // If the sum is equal to x, print the pair
    if (sum == x) {
        printf("%d\n", arr[low]);
        printf("%d\n", arr[high]);
        return;
    }

    // If the sum is less than x, move the low pointer to the right
    if (sum < x) {
        findPair(arr, low + 1, high, x);
    } else {
        // If the sum is greater than x, move the high pointer to the left
        findPair(arr, low, high - 1, x);
    }
}

int main() {
    int n;
    // Input size of the array
    scanf("%d", &n);

    int arr[n];

    // Input array elements
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }

    int x;
```


4-Two Elements sum to x: Atter
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245084&cmid=2656
// Input array elements
for (int i = 0; i < n; i++) {
scanf("%d", &arr[i]);
}
int x;
// Input the value of x
scanf("%d", &x);
// Call the recursive function to find the pair
findPair(arr, 0, n - 1, x);
return 0;
}
Open course index
Quiz navigation
1
Finish review
Input
Expected
Got
4
2
4
8
10
14
4
10
4
10
5
2
4
6
No
No
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:12
15-11-2025

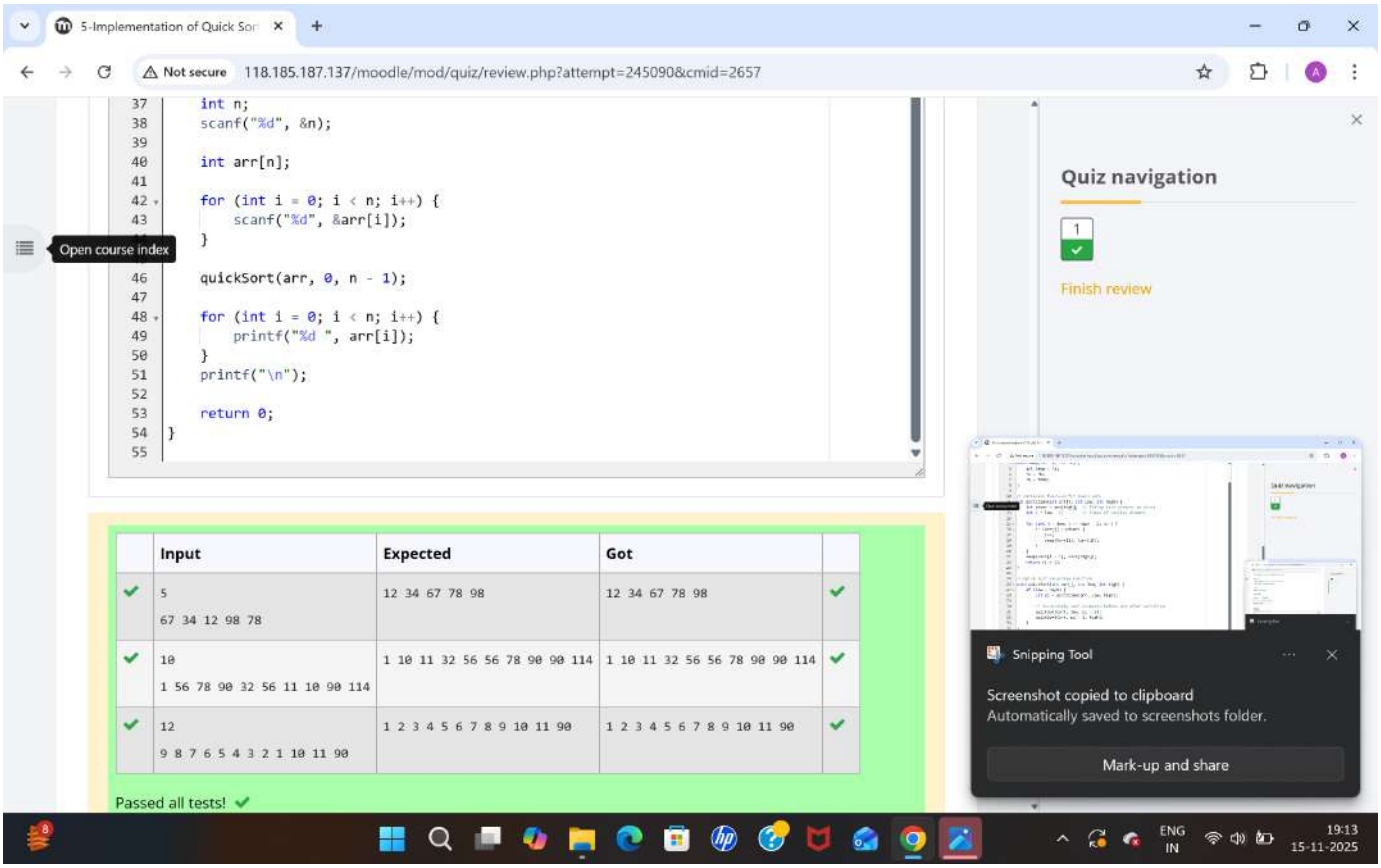

4-Two Elements sum to x: Atten
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245084&cmid=2656
Open course index
No
No
Passed all tests!
Correct
Marks for this submission: 1.00/1.00.
Finish review
Back to Course
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Data retention summary
ENG
IN
19:12
15-11-2025


CS23331-DAA-2024-CSE: 5-Imp
Not secure
118.185.187.137/moodle/mod/quiz/view.php?id=2657
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
Greedy Algorithms
1-G-Coin Problem
2-G-Cookies Problem
3-G-Burger Problem
4-G-Array Sum max problem
5-G-Product of Array elemen...
Dynamic Programming
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 5-Implementation of Quick Sort
5-Implementation of Quick Sort
Mark as done
Opened: Thursday, 30 May 2024, 9:49 AM
Re-attempt quiz
Attempts allowed: 2
Grading method: Highest grade
Summary of your previous attempts
ENG
IN
19:12
15-11-2025

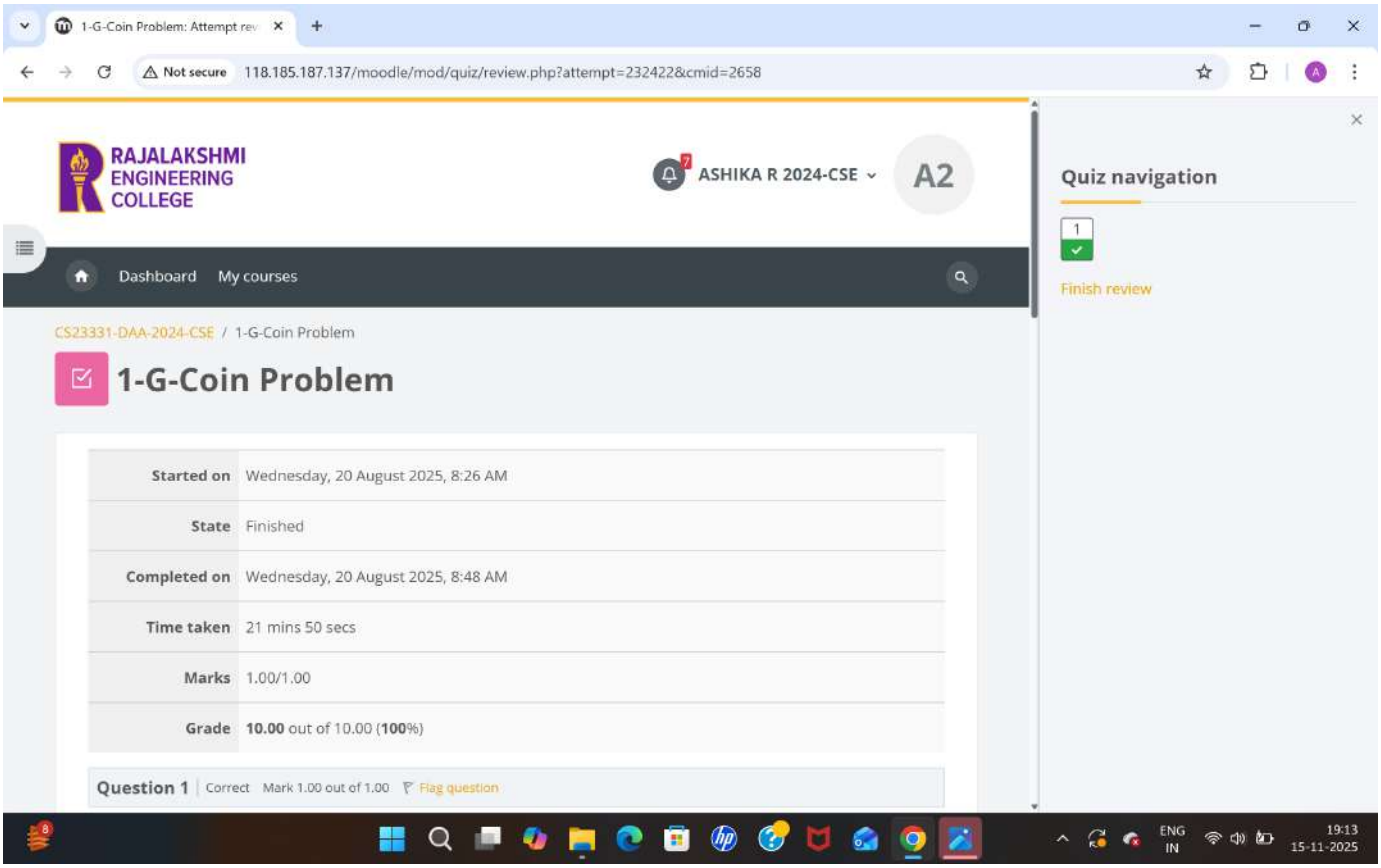

5-Implementation of Quick Sor
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=245090&cmid=2657
Question 1
Correct
Mark 1.00 out of 1.00
Flag question
Write a Program to Implement the Quick Sort Algorithm
Input Format:
The first line contains the no of elements in the list-n
The next n lines contain the elements.
Output:
Sorted list of elements
For example:
Input
Result
5
67 34 12 98 78
12 34 67 78 98
Answer:
#include <stdio.h>
// Function to swap two elements
void swap(int *a, int *b) {
int temp = *a;
*a = *b;
*b = temp;
}
Quiz navigation
1
Finish review
ENG
IN
19:12
15-11-2025

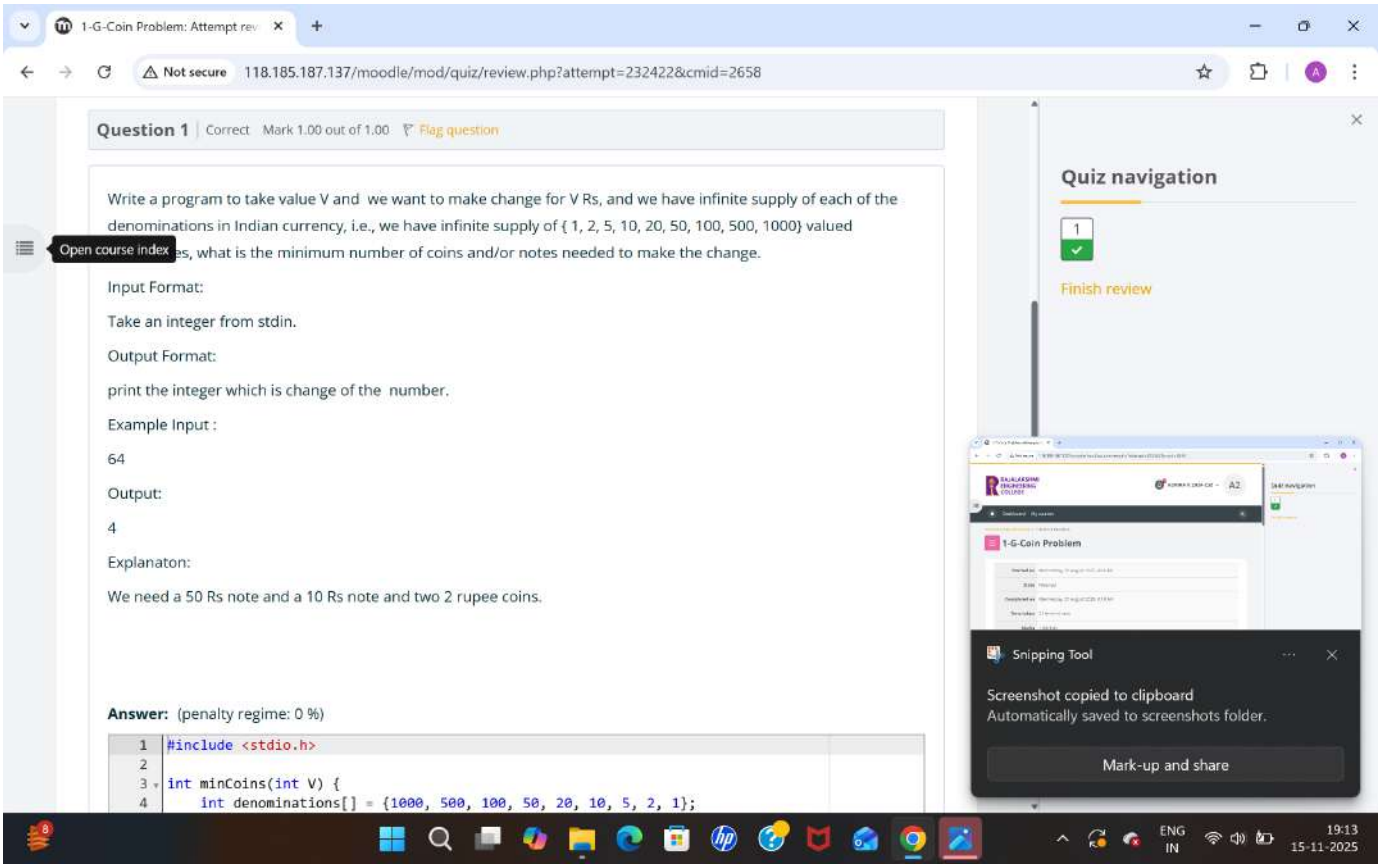


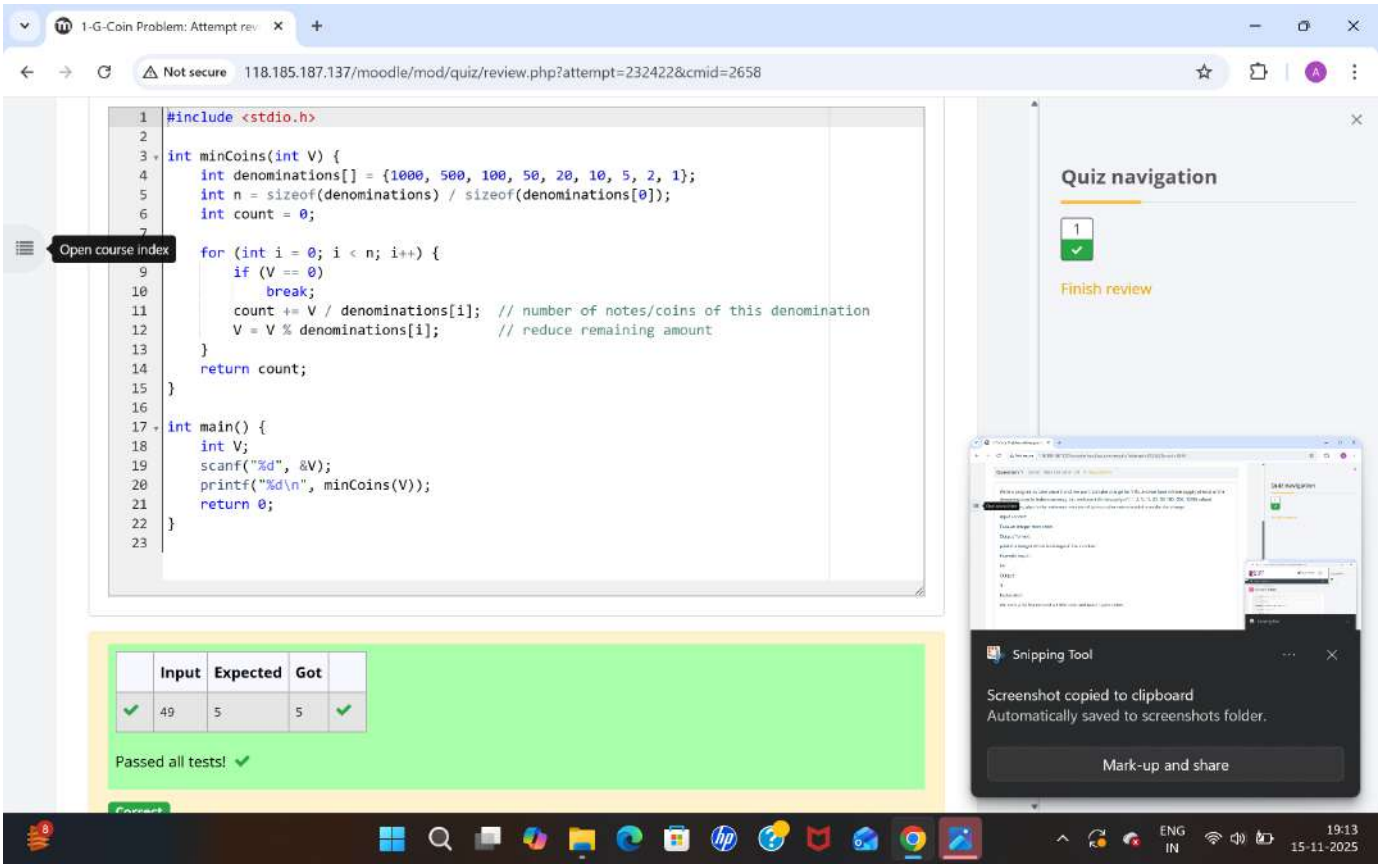

```c
void swap(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}

// Partition function for Quick Sort
int partition(int arr[], int low, int high) {
    int pivot = arr[high];  // Taking last element as pivot
    int i = low - 1;        // Index of smaller element

    for (int j = low; j <= high - 1; j++) {
        if (arr[j] < pivot) {
            i++;
            swap(&arr[i], &arr[j]);
        }
    }
    swap(&arr[i + 1], &arr[high]);
    return (i + 1);
}

// Quick Sort recursive function
void quickSort(int arr[], int low, int high) {
    if (low < high) {
        int pi = partition(arr, low, high);

        // Recursively sort elements before and after partition
        quickSort(arr, low, pi - 1);
        quickSort(arr, pi + 1, high);
    }
}

int main() {
    int n;
    scanf("%d", &n);

    int arr[n];
```

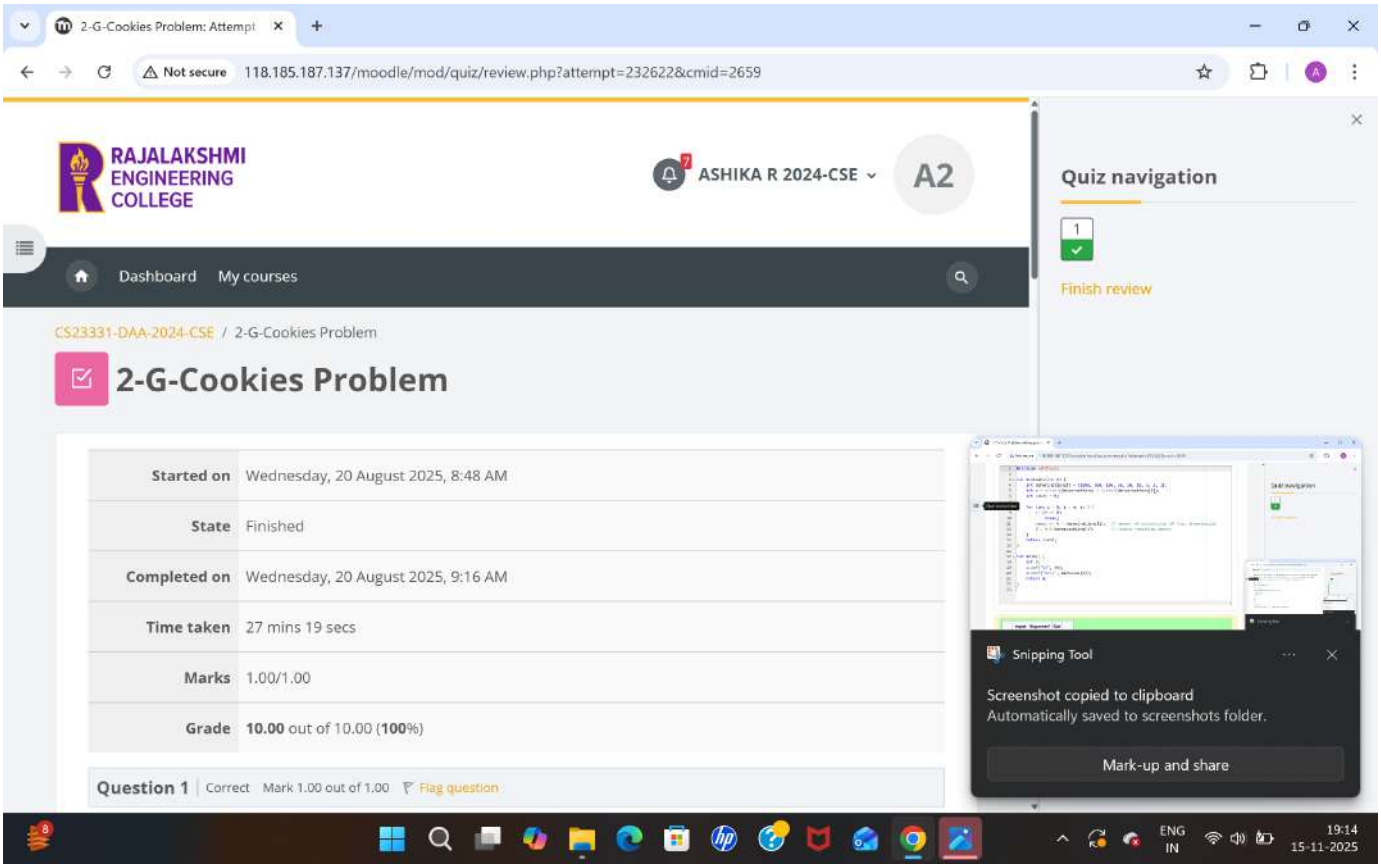


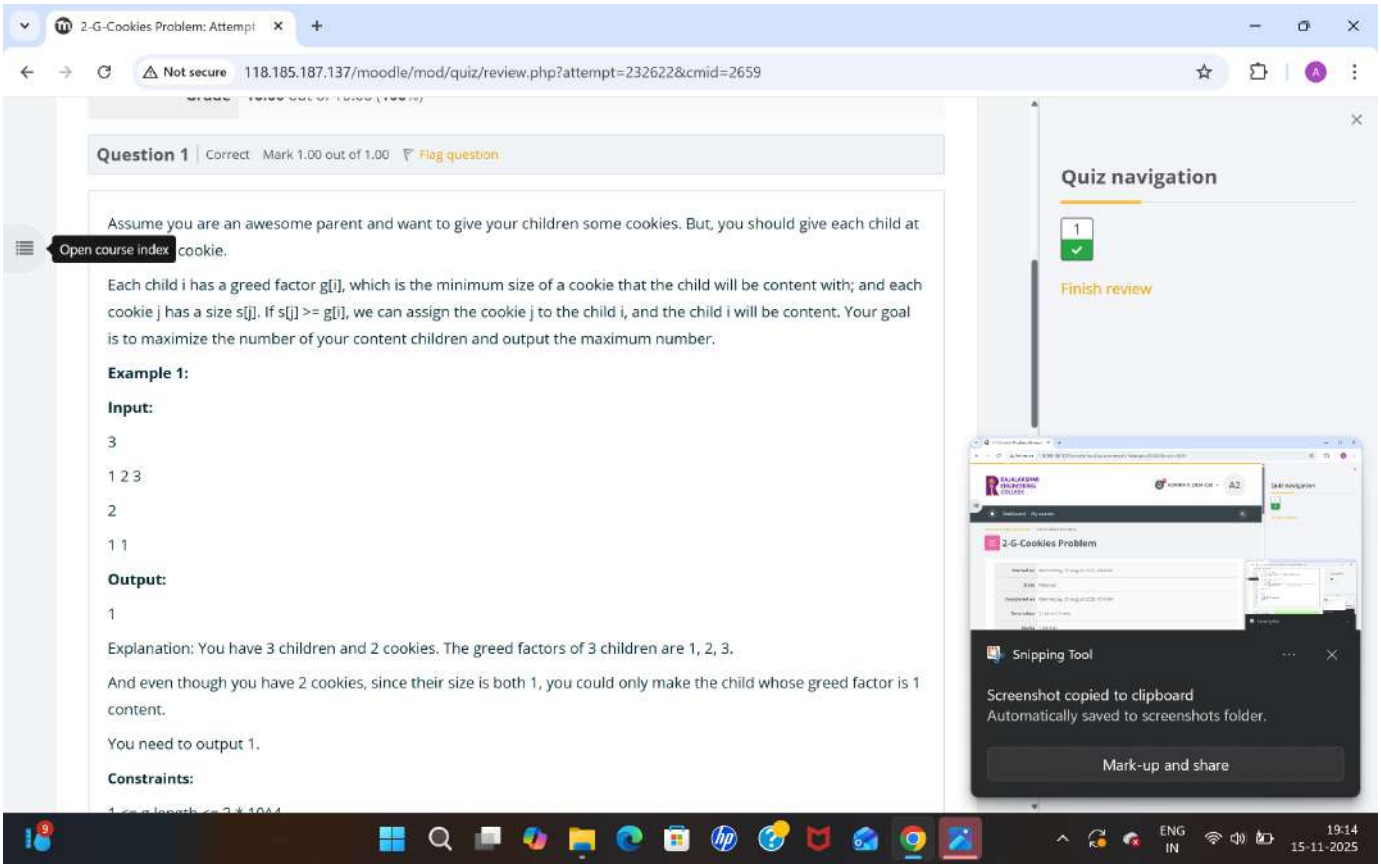

```c
    int n;
    scanf("%d", &n);

    int arr[n];

    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }

    quickSort(arr, 0, n - 1);

    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    printf("\n");

    return 0;
}
```


1-G-Coin Problem: Attempt rev
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=232422&cmid=2658
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 1-G-Coin Problem
1-G-Coin Problem
Started on Wednesday, 20 August 2025, 8:26 AM
State Finished
Completed on Wednesday, 20 August 2025, 8:48 AM
Time taken 21 mins 50 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
ENG IN
19:13
15-11-2025


1-G-Coin Problem: Attempt rev
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=232422&cmid=2658
Question 1
Correct
Mark 1.00 out of 1.00
Flag question
Open course index
Write a program to take value V and we want to make change for V Rs, and we have infinite supply of each of the denominations in Indian currency, i.e., we have infinite supply of { 1, 2, 5, 10, 20, 50, 100, 500, 1000} valued
es, what is the minimum number of coins and/or notes needed to make the change.
Input Format:
Take an integer from stdin.
Output Format:
print the integer which is change of the number.
Example Input :
64
Output:
4
Explanaton:
We need a 50 Rs note and a 10 Rs note and two 2 rupee coins.
Answer: (penalty regime: 0 %)
#include <stdio.h>
int minCoins(int V) {
int denominations[] = {1000, 500, 100, 50, 20, 10, 5, 2, 1};
Quiz navigation
1
Finish review
1-G-Coin Problem
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:13
15-11-2025


1-G-Coin Problem: Attempt rev
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=232422&cmid=2658
Open course index
#include <stdio.h>
int minCoins(int V) {
int denominations[] = {1000, 500, 100, 50, 20, 10, 5, 2, 1};
int n = sizeof(denominations) / sizeof(denominations[0]);
int count = 0;
for (int i = 0; i < n; i++) {
if (V == 0)
break;
count += V / denominations[i]; // number of notes/coins of this denomination
V = V % denominations[i]; // reduce remaining amount
}
return count;
}
int main() {
int V;
scanf("%d", &V);
printf("%d\n", minCoins(V));
return 0;
}
Input
Expected
Got
49
5
5
Passed all tests!
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:13
15-11-2025


2-G-Cookies Problem: Attempt
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=232622&cmid=2659
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 2-G-Cookies Problem
2-G-Cookies Problem
Started on Wednesday, 20 August 2025, 8:48 AM
State Finished
Completed on Wednesday, 20 August 2025, 9:16 AM
Time taken 27 mins 19 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:14
15-11-2025


2-G-Cookies Problem: Attempt
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=232622&cmid=2659
Question 1
Correct
Mark 1.00 out of 1.00
Flag question
Open course index
Assume you are an awesome parent and want to give your children some cookies. But, you should give each child at most one cookie.
Each child i has a greed factor g[i], which is the minimum size of a cookie that the child will be content with; and each cookie j has a size s[j]. If s[j] >= g[i], we can assign the cookie j to the child i, and the child i will be content. Your goal is to maximize the number of your content children and output the maximum number.
Example 1:
Input:
3
1 2 3
2
1 1
Output:
1
Explanation: You have 3 children and 2 cookies. The greed factors of 3 children are 1, 2, 3.
And even though you have 2 cookies, since their size is both 1, you could only make the child whose greed factor is 1 content.
You need to output 1.
Constraints:
Quiz navigation
1
Finish review
2-G-Cookies Problem
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:14
15-11-2025

2-G-Cookies Problem: Attempt

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=232622&cmid=2659

1 <= g.length <= 3 * 10^4

0 <= s.length <= 3 * 10^4

1 <= g[i], s[j] <= 2^31 - 1

Open course index

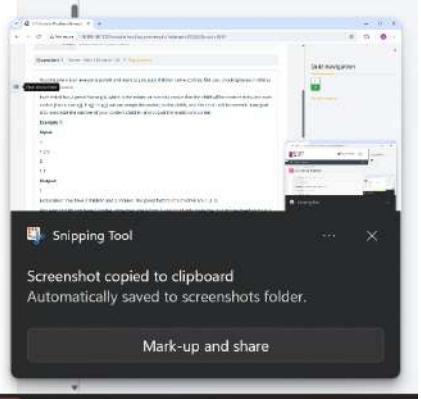
Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <stdlib.h>

int cmp(const void *a, const void *b) {
    return (*(int*)a - *(int*)b);
}

int main() {
    int n, m;
    scanf("%d", &n);
    int g[n];
    for (int i = 0; i < n; i++) scanf("%d", &g[i]);

    scanf("%d", &m);
    int s[m];
    for (int i = 0; i < m; i++) scanf("%d", &s[i]);

    qsort(g, n, sizeof(int), cmp);
    qsort(s, m, sizeof(int), cmp);

    int i = 0, j = 0, count = 0;
    while (i < n && j < m) {
        if (s[j] >= g[i]) {
            count++; i++; j++;
        } else {
            j++;
        }
```

Quiz navigation

1

Finish review



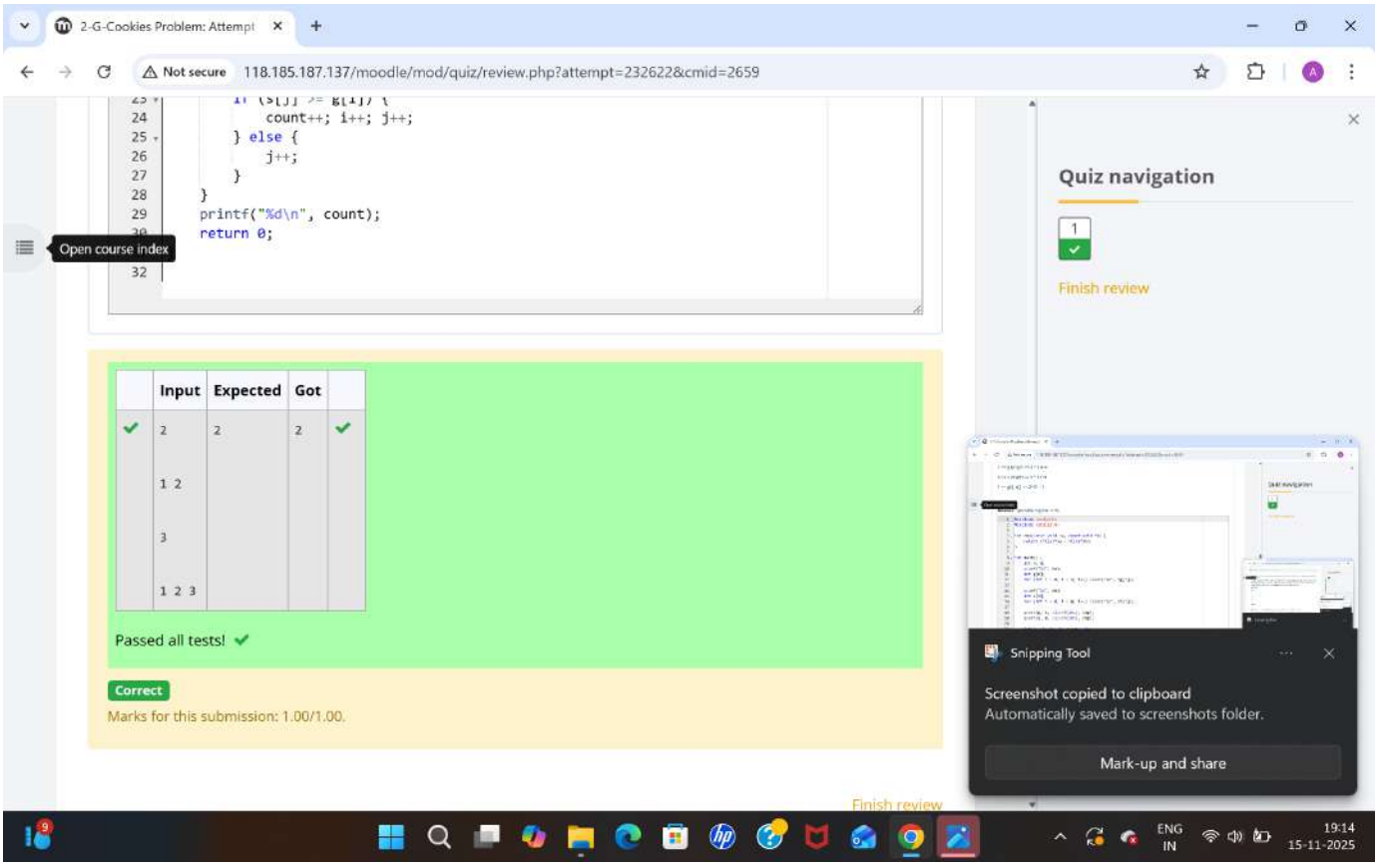
Snipping Tool

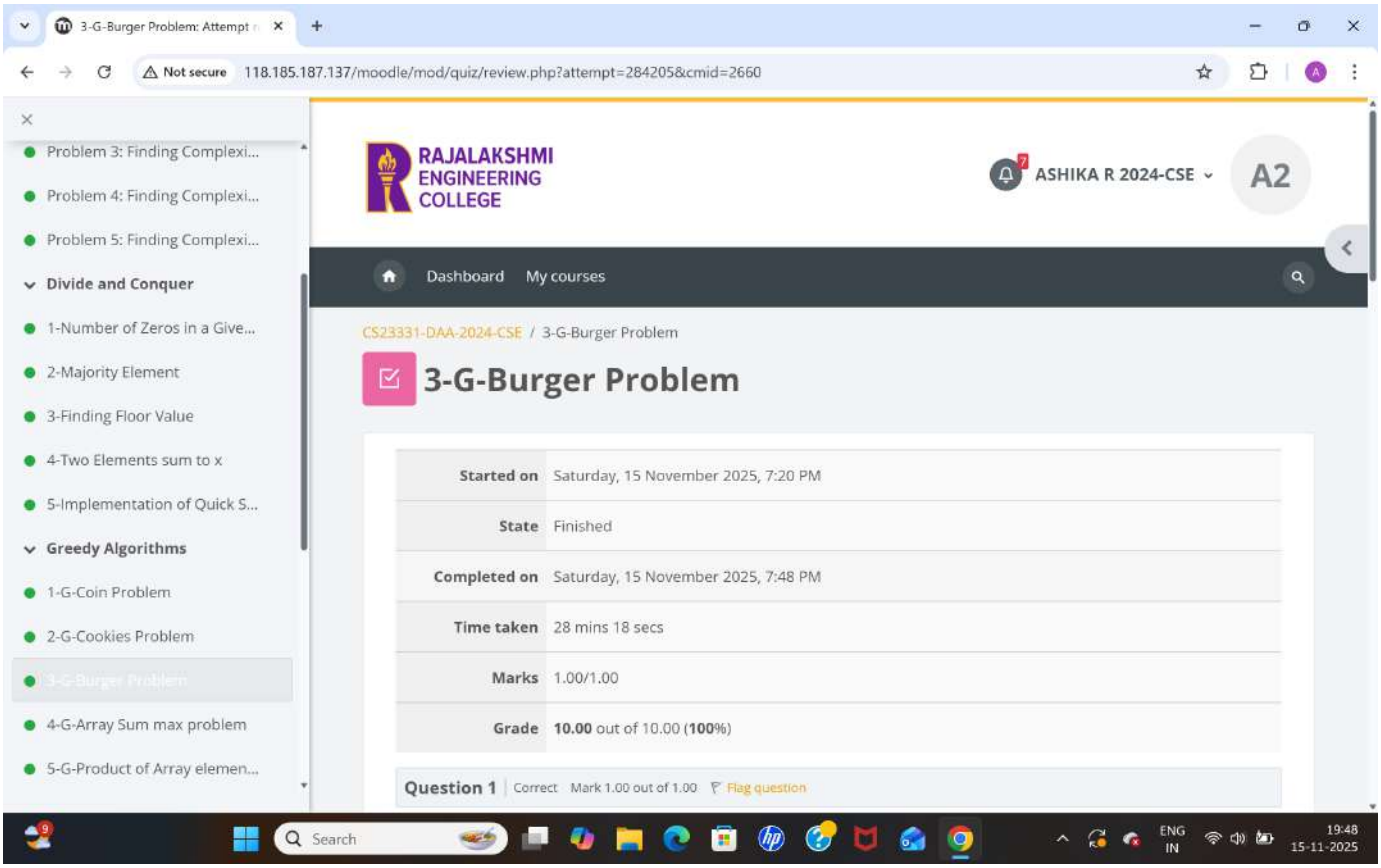
Screenshot copied to clipboard
Automatically saved to screenshots folder.

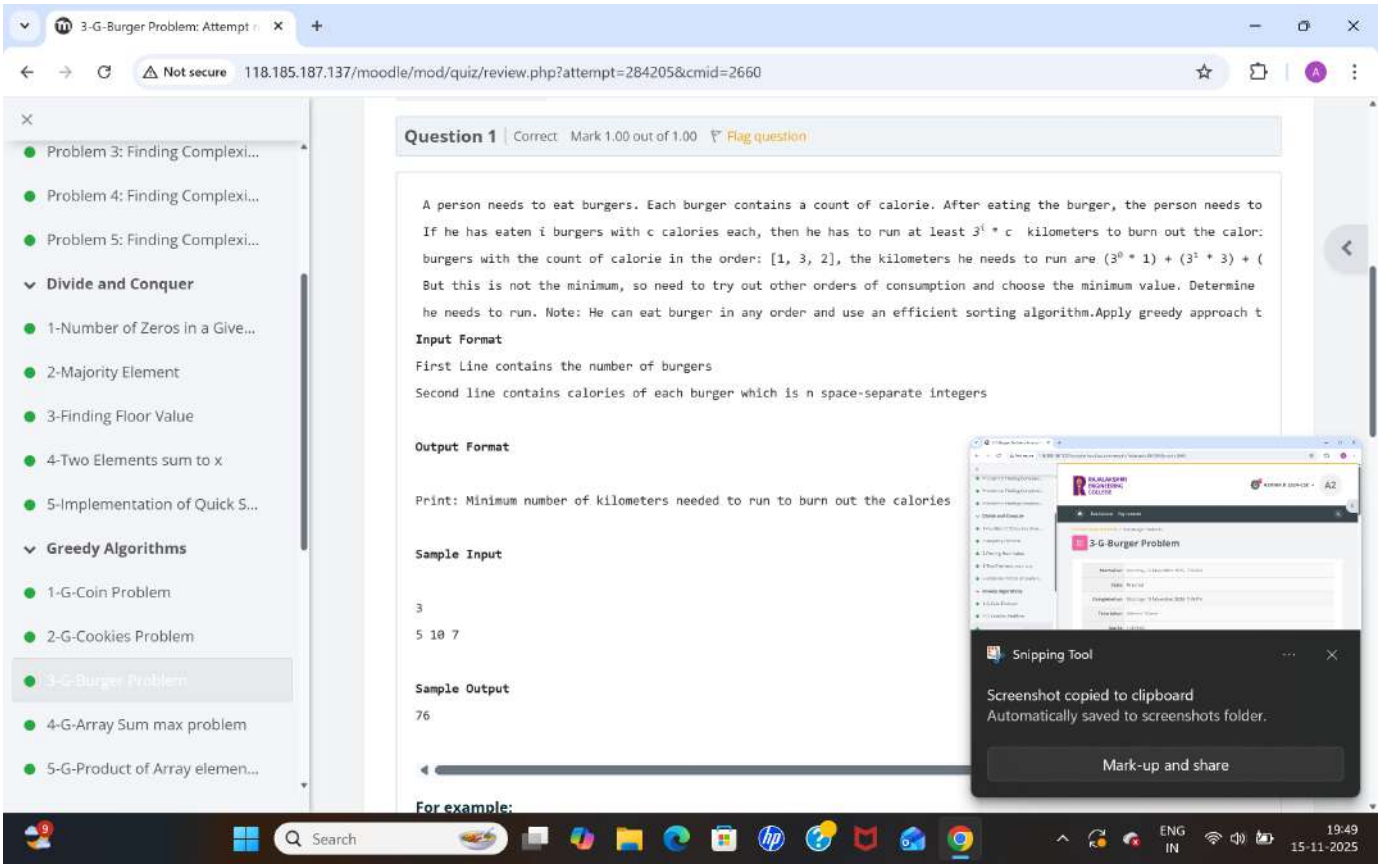
Mark-up and share

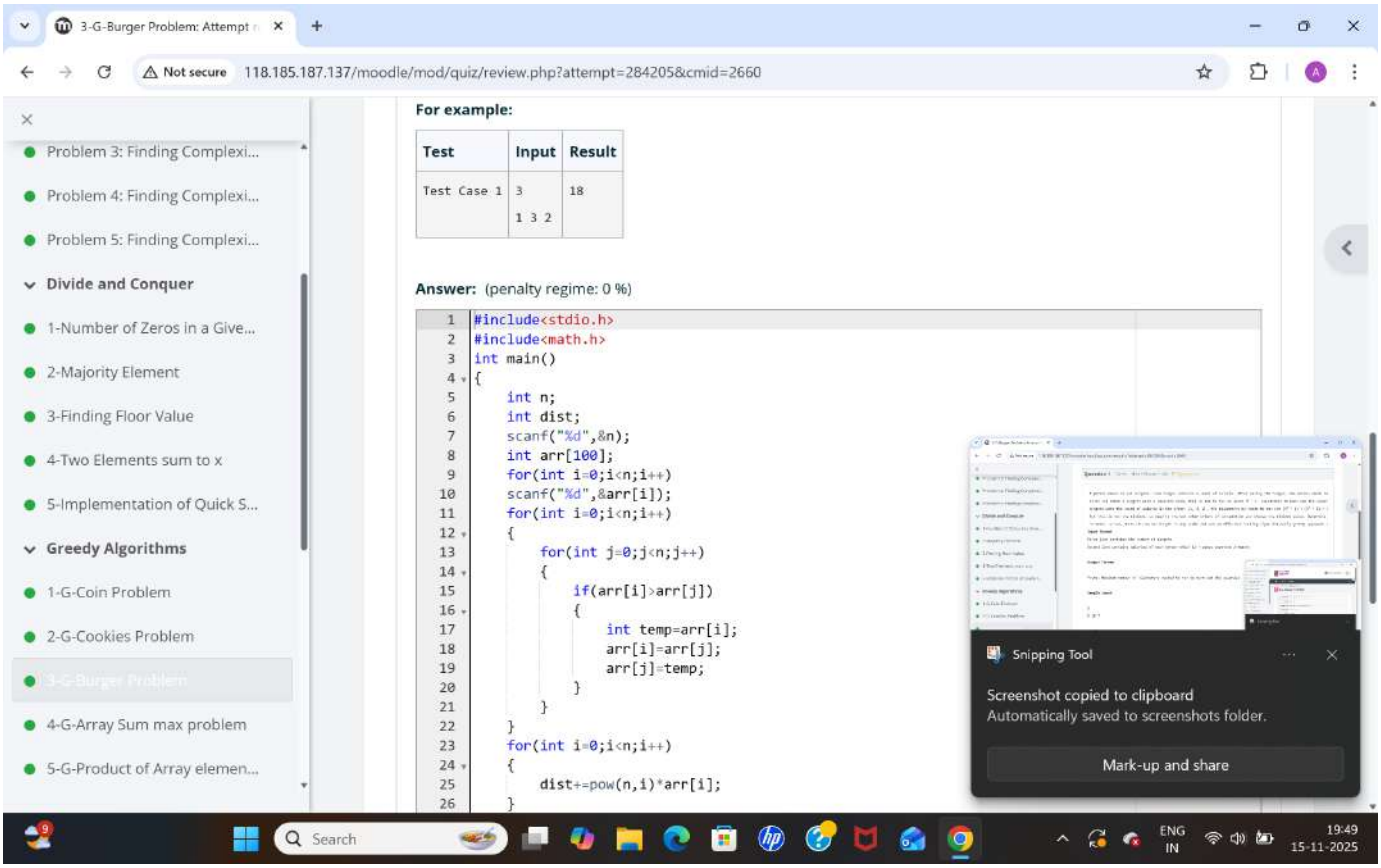
ENG
IN

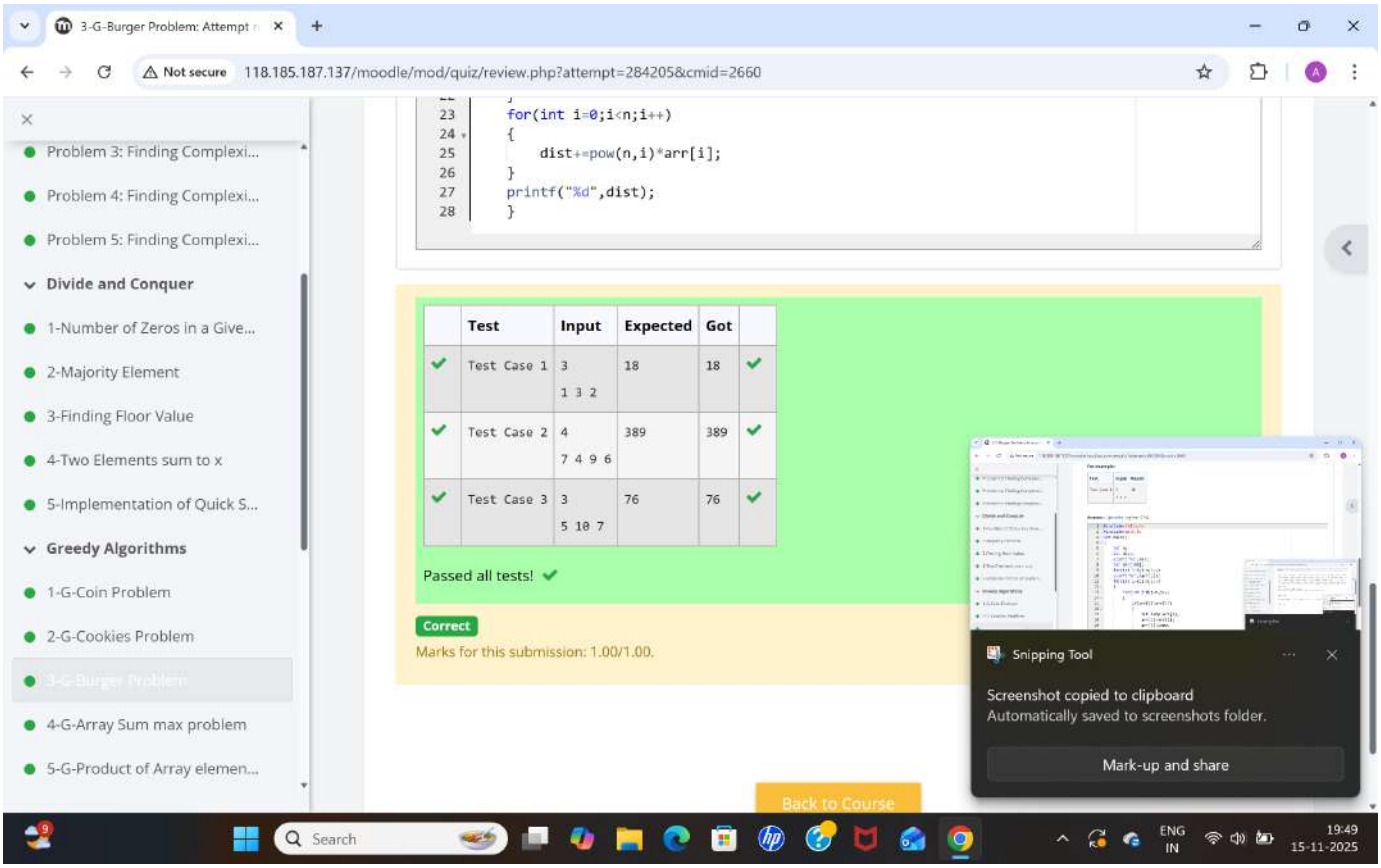
19:14
15-11-2025

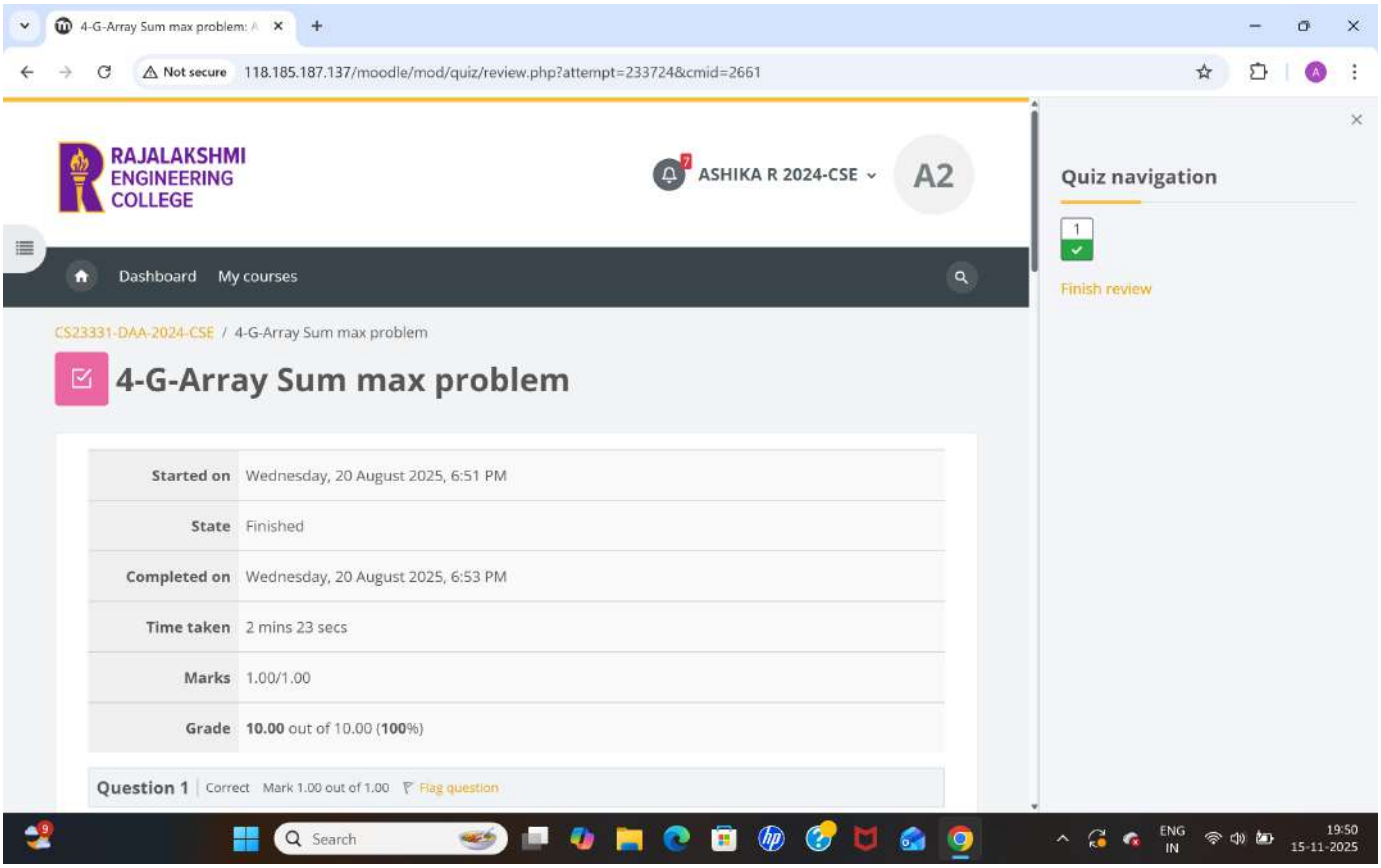

2-G-Cookies Problem: Attempt
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=232622&cmid=2659
Open course index
Quiz navigation
1
Finish review
Input
Expected
Got
2
1 2
3
1 2 3
2
2
Passed all tests!
Correct
Marks for this submission: 1.00/1.00.
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
ENG
IN
19:14
15-11-2025

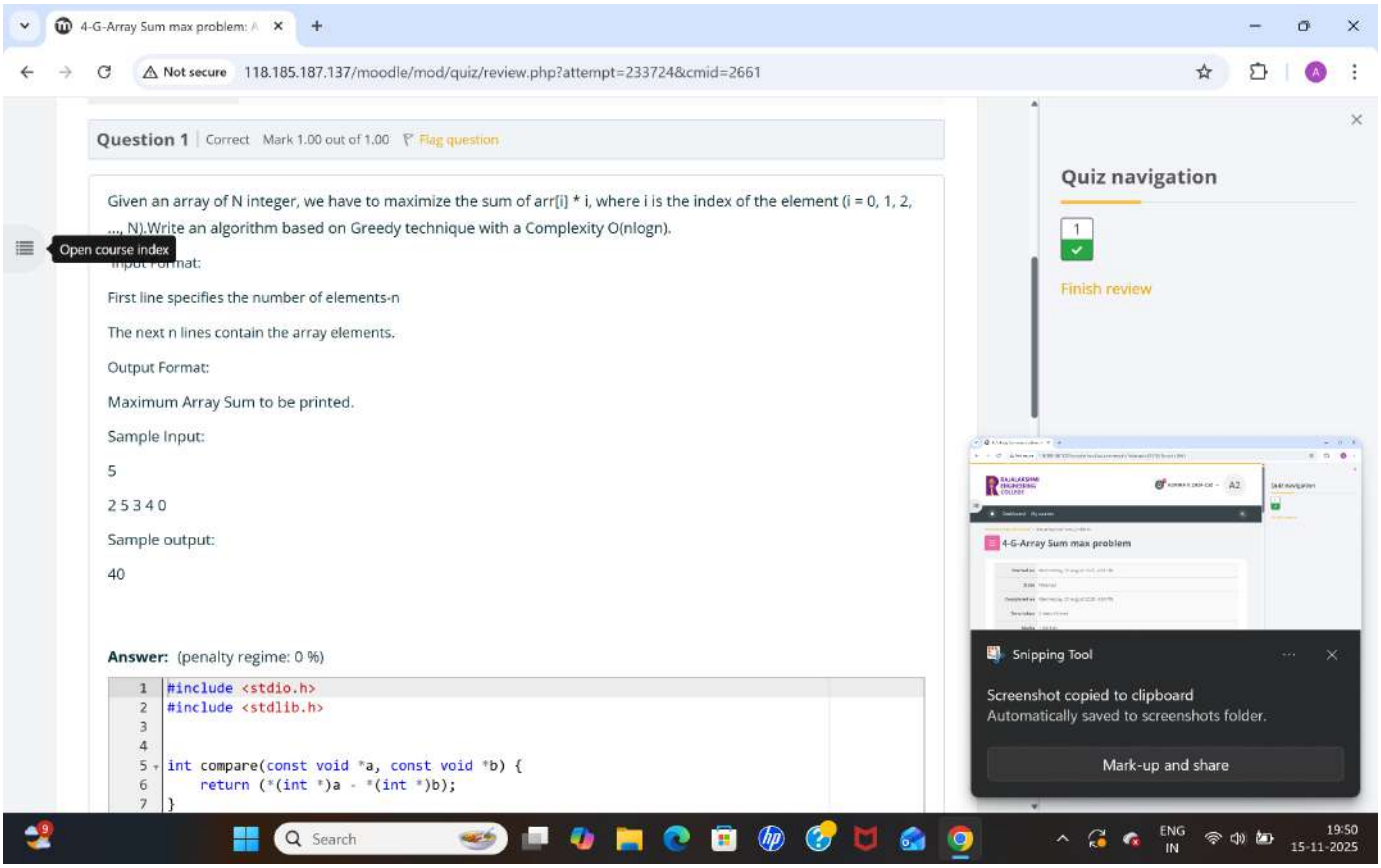

3-G-Burger Problem: Attempt
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=284205&cmid=2660
Problem 3: Finding Complexi...
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
Greedy Algorithms
1-G-Coin Problem
2-G-Cookies Problem
3-G-Burger Problem
4-G-Array Sum max problem
5-G-Product of Array elemen...
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 3-G-Burger Problem
3-G-Burger Problem
Started on Saturday, 15 November 2025, 7:20 PM
State Finished
Completed on Saturday, 15 November 2025, 7:48 PM
Time taken 28 mins 18 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Search
ENG IN
19:48
15-11-2025

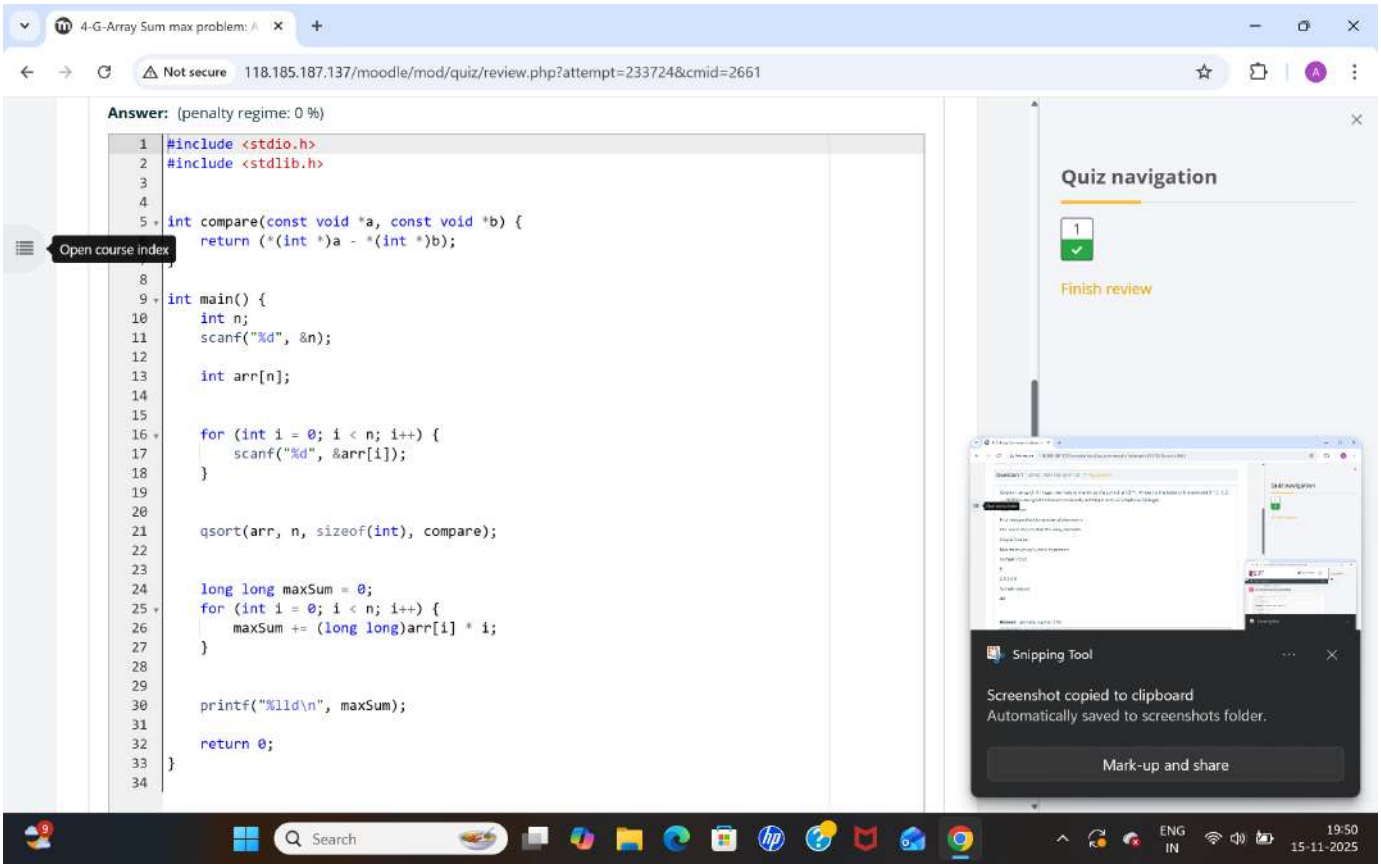

3-G-Burger Problem: Attempt
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=284205&cmid=2660
Problem 3: Finding Complexi...
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
Greedy Algorithms
1-G-Coin Problem
2-G-Cookies Problem
3-G-Burger Problem
4-G-Array Sum max problem
5-G-Product of Array elemen...
Question 1
Correct
Mark 1.00 out of 1.00
Flag question
A person needs to eat burgers. Each burger contains a count of calorie. After eating the burger, the person needs to
If he has eaten i burgers with c calories each, then he has to run at least 3^i * c kilometers to burn out the calor
burgers with the count of calorie in the order: [1, 3, 2], the kilometers he needs to run are (3^0 * 1) + (3^1 * 3) + (
But this is not the minimum, so need to try out other orders of consumption and choose the minimum value. Determine
he needs to run. Note: He can eat burger in any order and use an efficient sorting algorithm.Apply greedy approach t
Input Format
First Line contains the number of burgers
Second line contains calories of each burger which is n space-separate integers
Output Format
Print: Minimum number of kilometers needed to run to burn out the calories
Sample Input
3
5 10 7
Sample Output
76
For example:
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:49
15-11-2025

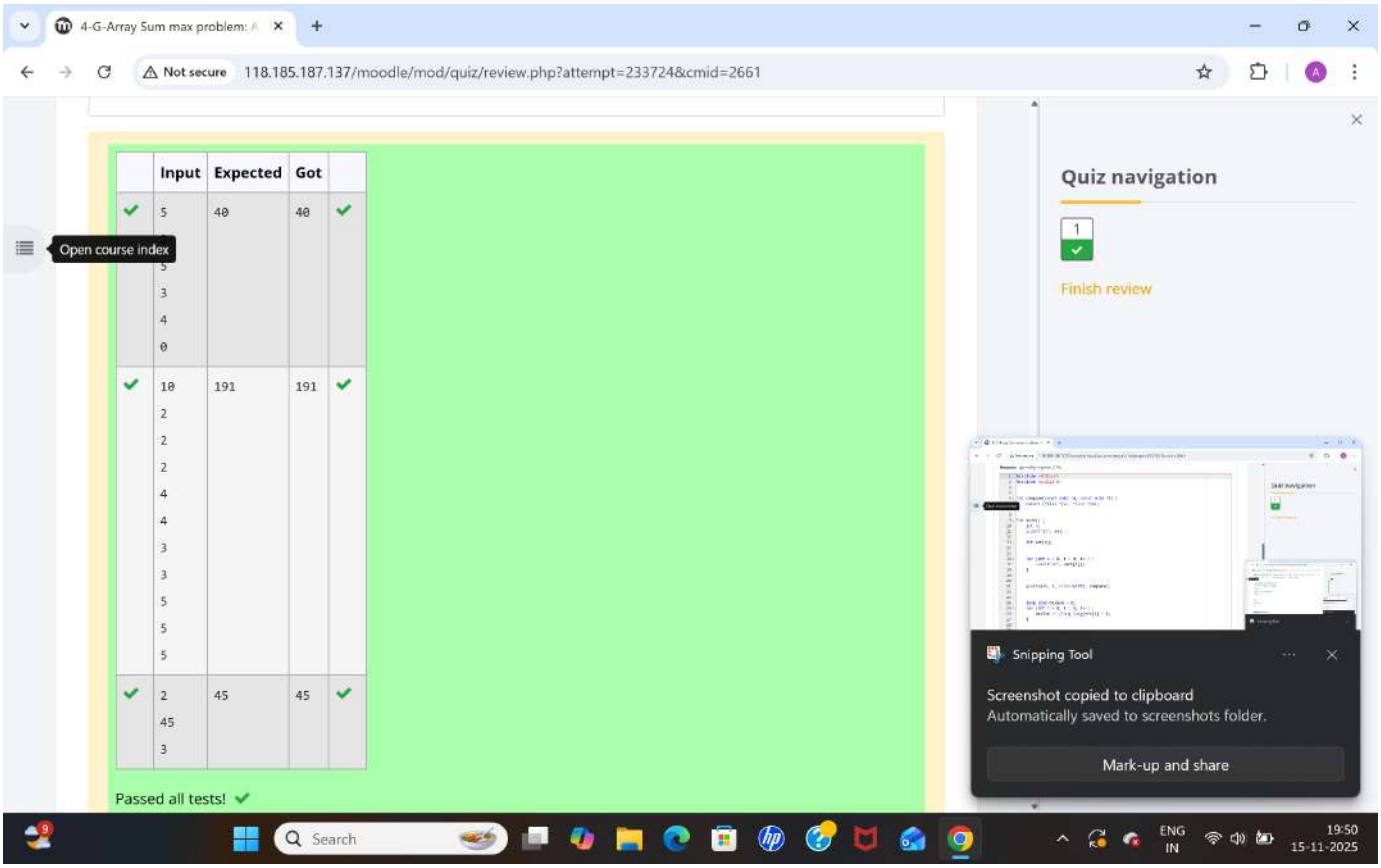

3-G-Burger Problem: Attempt
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=284205&cmid=2660
Problem 3: Finding Complexi...
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
Greedy Algorithms
1-G-Coin Problem
2-G-Cookies Problem
3-G-Burger Problem
4-G-Array Sum max problem
5-G-Product of Array elemen...
For example:
Test
Input
Result
Test Case 1
3
1 3 2
18
Answer: (penalty regime: 0 %)
#include<stdio.h>
#include<math.h>
int main()
{
int n;
int dist;
scanf("%d",&n);
int arr[100];
for(int i=0;i<n;i++)
scanf("%d",&arr[i]);
for(int i=0;i<n;i++)
{
for(int j=0;j<n;j++)
{
if(arr[i]>arr[j])
{
int temp=arr[i];
arr[i]=arr[j];
arr[j]=temp;
}
}
}
for(int i=0;i<n;i++)
{
dist+=pow(n,i)*arr[i];
}
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:49
15-11-2025

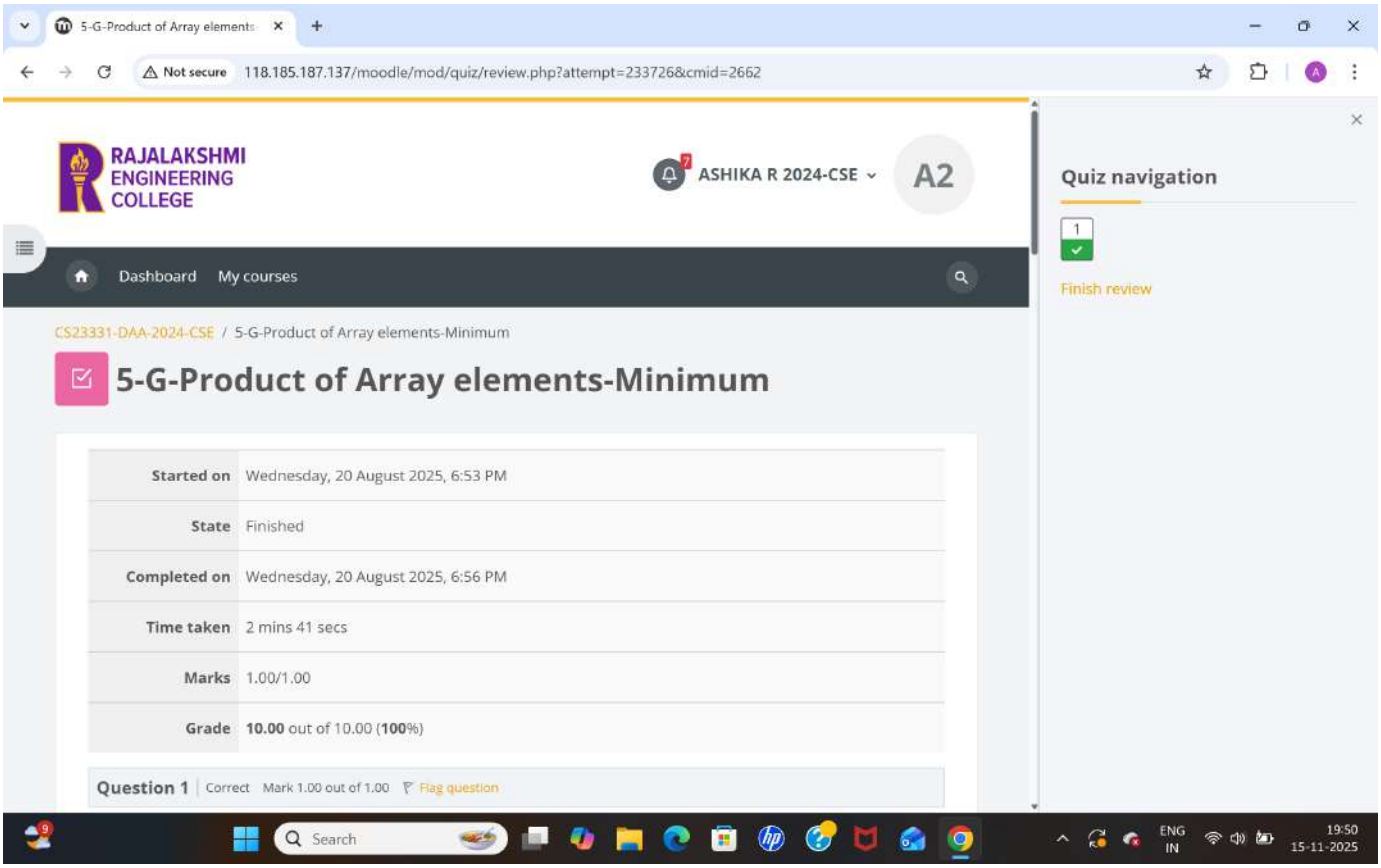

3-G-Burger Problem: Attempt
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=284205&cmid=2660
Problem 3: Finding Complexi...
Problem 4: Finding Complexi...
Problem 5: Finding Complexi...
Divide and Conquer
1-Number of Zeros in a Give...
2-Majority Element
3-Finding Floor Value
4-Two Elements sum to x
5-Implementation of Quick S...
Greedy Algorithms
1-G-Coin Problem
2-G-Cookies Problem
3-G-Burger Problem
4-G-Array Sum max problem
5-G-Product of Array elemen...
for(int i=0;i<n;i++)
{
dist+=pow(n,i)*arr[i];
}
printf("%d",dist);
}
Test
Input
Expected
Got
Test Case 1
3
1 3 2
18
18
Test Case 2
4
7 4 9 6
389
389
Test Case 3
3
5 10 7
76
76
Passed all tests!
Correct
Marks for this submission: 1.00/1.00.
Back to Course
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:49
15-11-2025


4-G-Array Sum max problem: A
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233724&cmid=2661
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 4-G-Array Sum max problem
4-G-Array Sum max problem
Started on Wednesday, 20 August 2025, 6:51 PM
State Finished
Completed on Wednesday, 20 August 2025, 6:53 PM
Time taken 2 mins 23 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
Search
ENG IN
19:50
15-11-2025


4-G-Array Sum max problem: A
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233724&cmid=2661
Question 1
Correct
Mark 1.00 out of 1.00
Flag question
Given an array of N integer, we have to maximize the sum of arr[i] * i, where i is the index of the element (i = 0, 1, 2, ... N).Write an algorithm based on Greedy technique with a Complexity O(nlogn).
Open course index
Input Format:
First line specifies the number of elements-n
The next n lines contain the array elements.
Output Format:
Maximum Array Sum to be printed.
Sample Input:
5
2 5 3 4 0
Sample output:
40
Answer: (penalty regime: 0 %)
#include <stdio.h>
#include <stdlib.h>
int compare(const void *a, const void *b) {
return (*(int *)a - *(int *)b);
}
Quiz navigation
1
Finish review
4-G-Array Sum max problem
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:50
15-11-2025


4-G-Array Sum max problem:
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233724&cmid=2661
Answer: (penalty regime: 0 %)
Open course index
#include <stdio.h>
#include <stdlib.h>
int compare(const void *a, const void *b) {
return (*(int *)a - *(int *)b);
}
int main() {
int n;
scanf("%d", &n);
int arr[n];
for (int i = 0; i < n; i++) {
scanf("%d", &arr[i]);
}
qsort(arr, n, sizeof(int), compare);
long long maxSum = 0;
for (int i = 0; i < n; i++) {
maxSum += (long long)arr[i] * i;
}
printf("%lld\n", maxSum);
return 0;
}
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:50
15-11-2025


4-G-Array Sum max problem:
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233724&cmid=2661
Open course index
Input
Expected
Got
5
5
3
4
0
40
40
10
2
2
2
4
4
3
3
5
5
5
191
191
2
45
3
45
45
Passed all tests!
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:50
15-11-2025


5-G-Product of Array elements
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233726&cmid=2662
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 5-G-Product of Array elements-Minimum
5-G-Product of Array elements-Minimum
Started on Wednesday, 20 August 2025, 6:53 PM
State Finished
Completed on Wednesday, 20 August 2025, 6:56 PM
Time taken 2 mins 41 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
Search
ENG IN
19:50
15-11-2025

5-G-Product of Array elements

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=233726&cmid=2662

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

Given two arrays array_One[] and array_Two[] of same size N. We need to first rearrange the arrays such that the sum of the product of pairs(1 element from each) is minimum. That is SUM (A[i] * B[i]) for all i is minimum.

Open course index

ple:

Input	Result
3 1 2 3 4 5 6	28

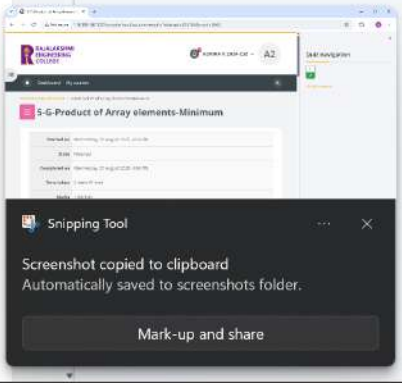
Answer: (penalty regime: 0 %)

```
#include <stdio.h>
#include <stdlib.h>


int compareAsc(const void *a, const void *b) {
    return (*(int *)a - *(int *)b);
}


int compareDesc(const void *a, const void *b) {
    return (*(int *)b - *(int *)a);
}
```

Quiz navigation

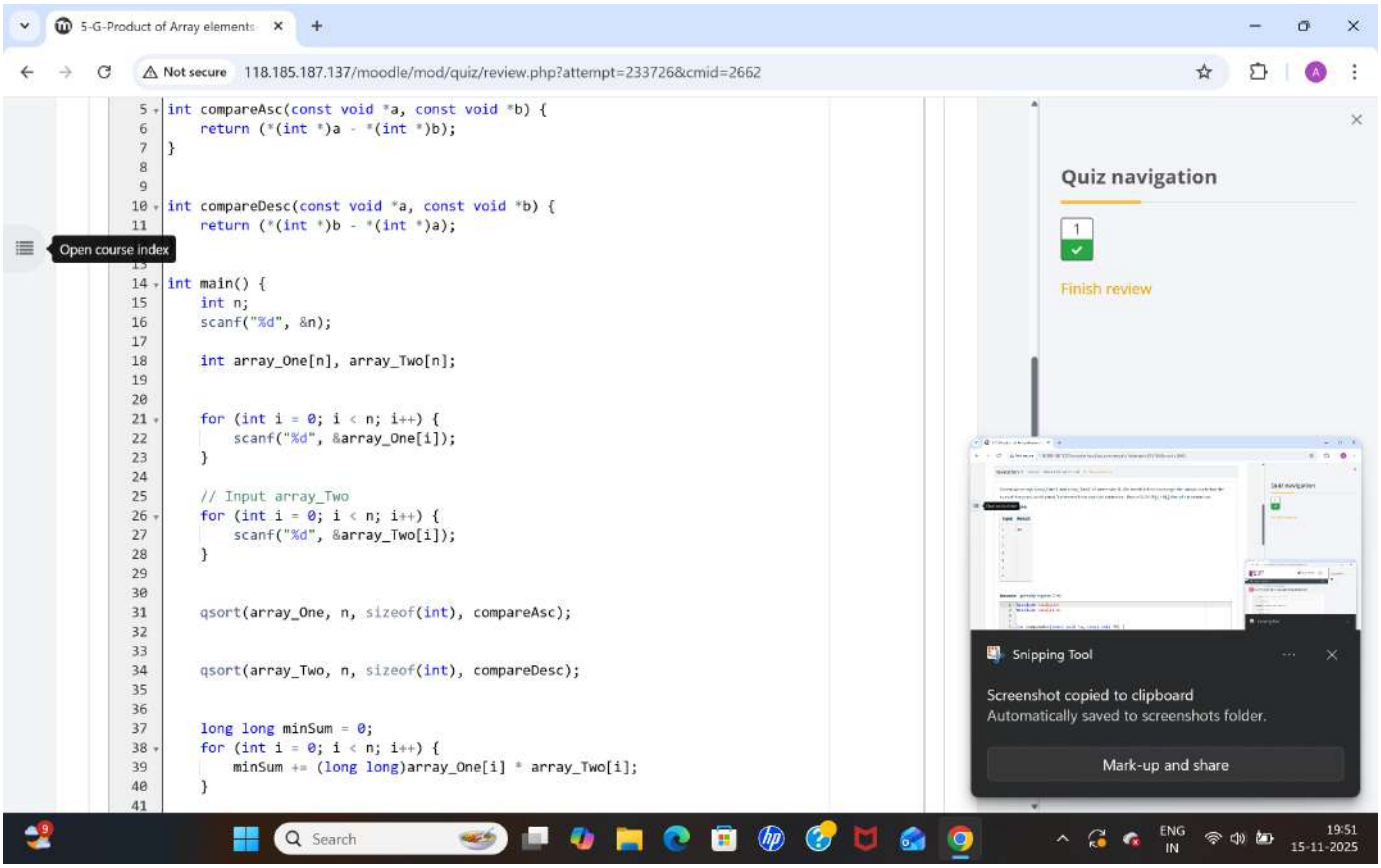
1

Finish review

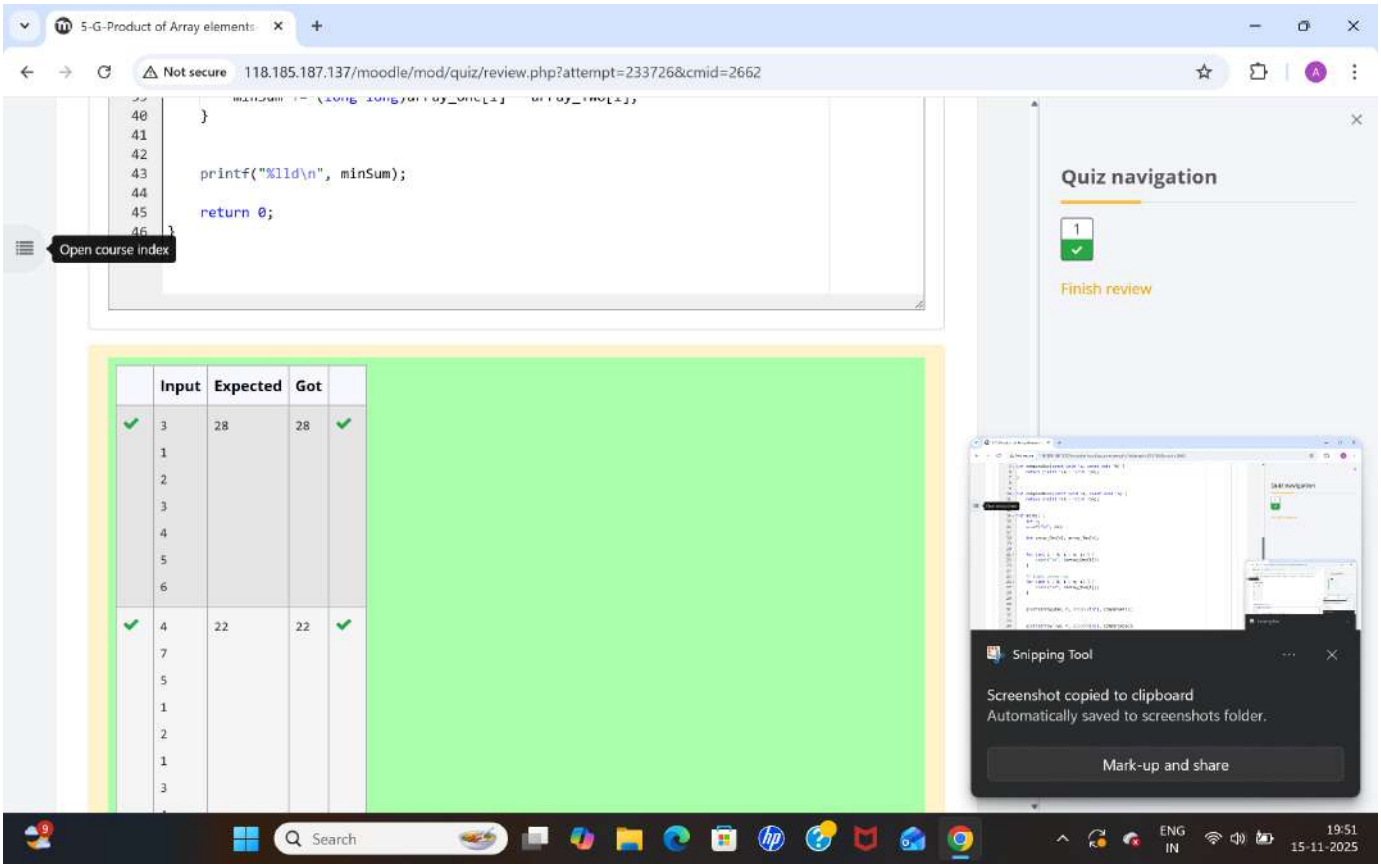


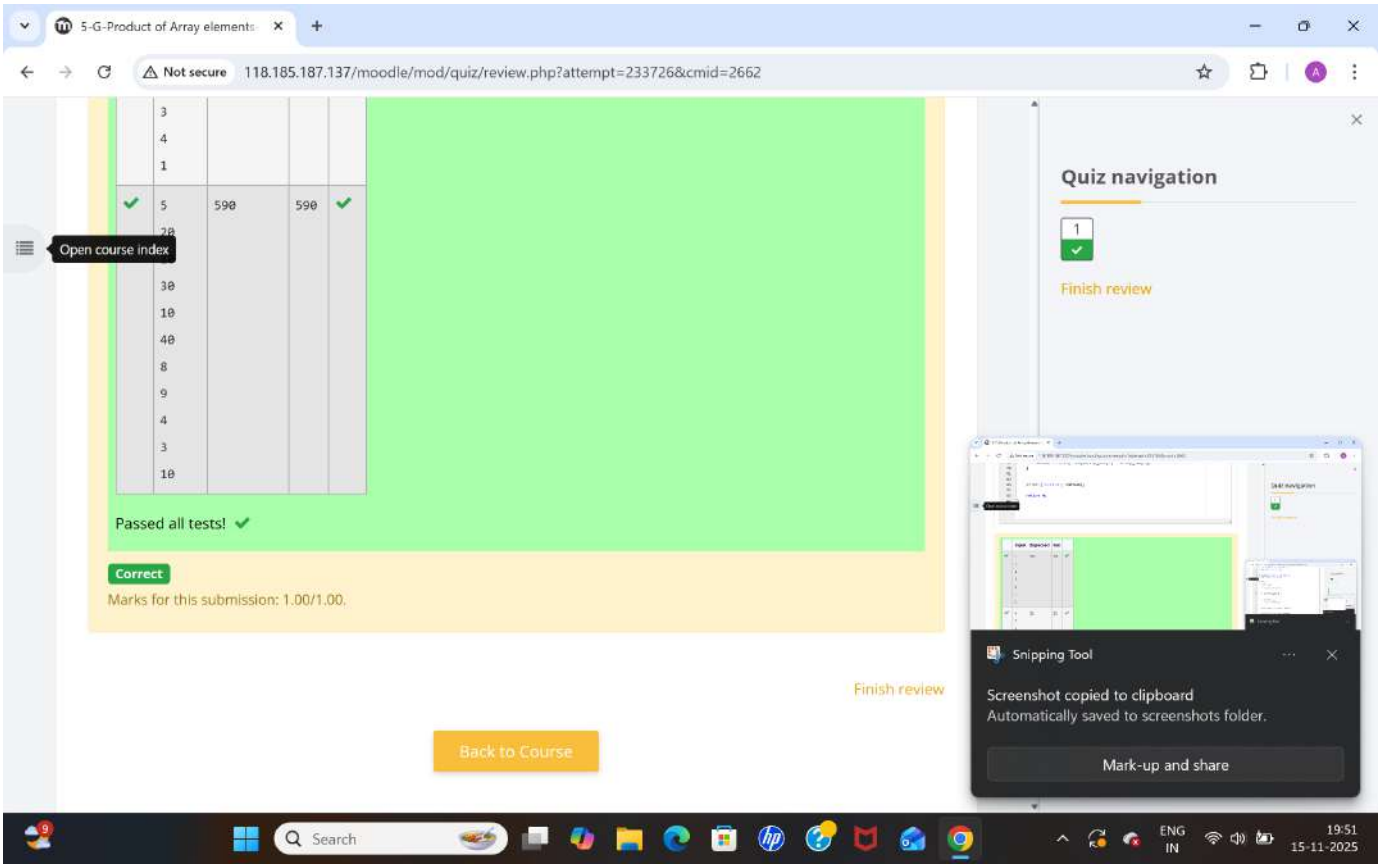
Search

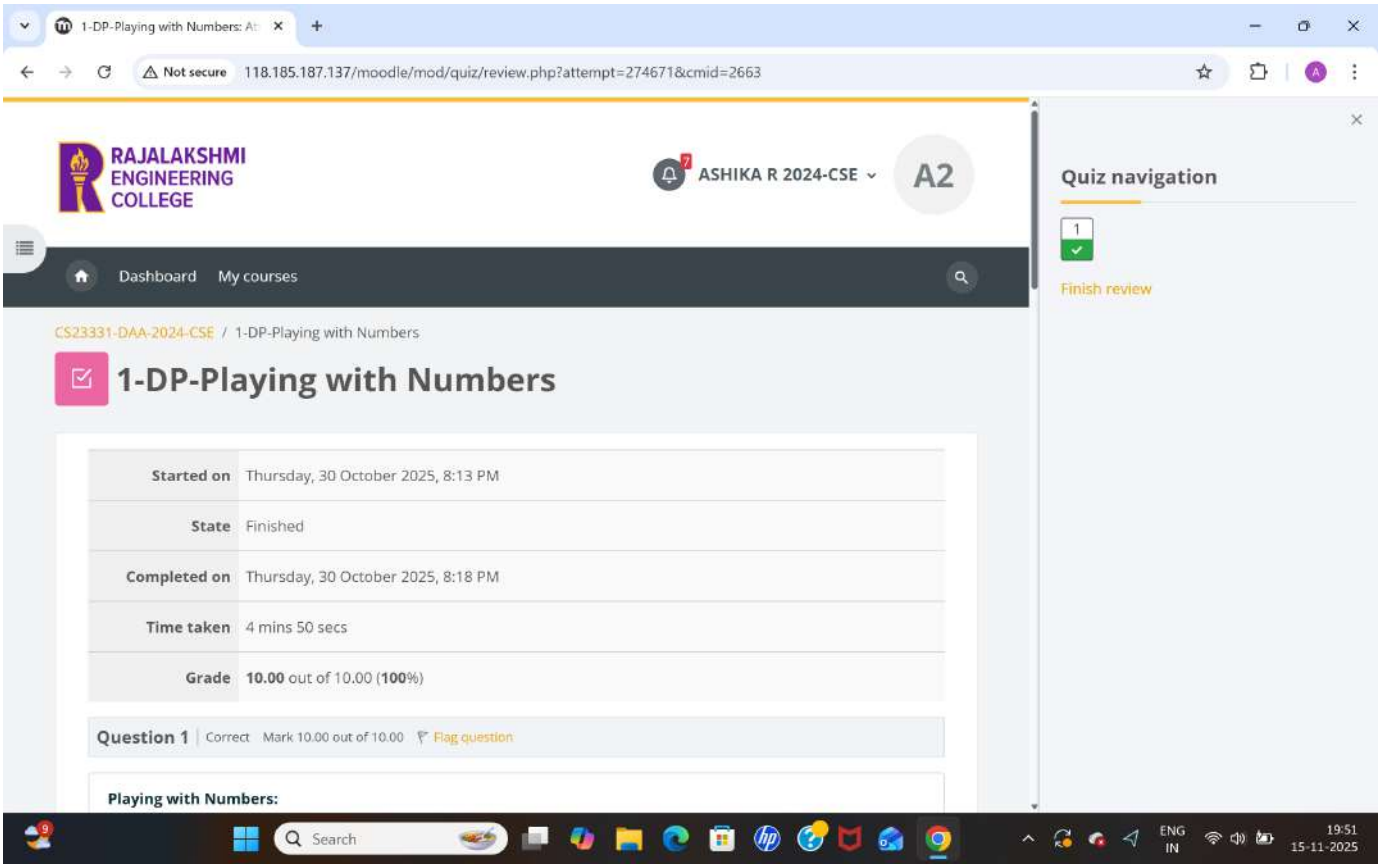
ENG IN 19:51 15-11-2025

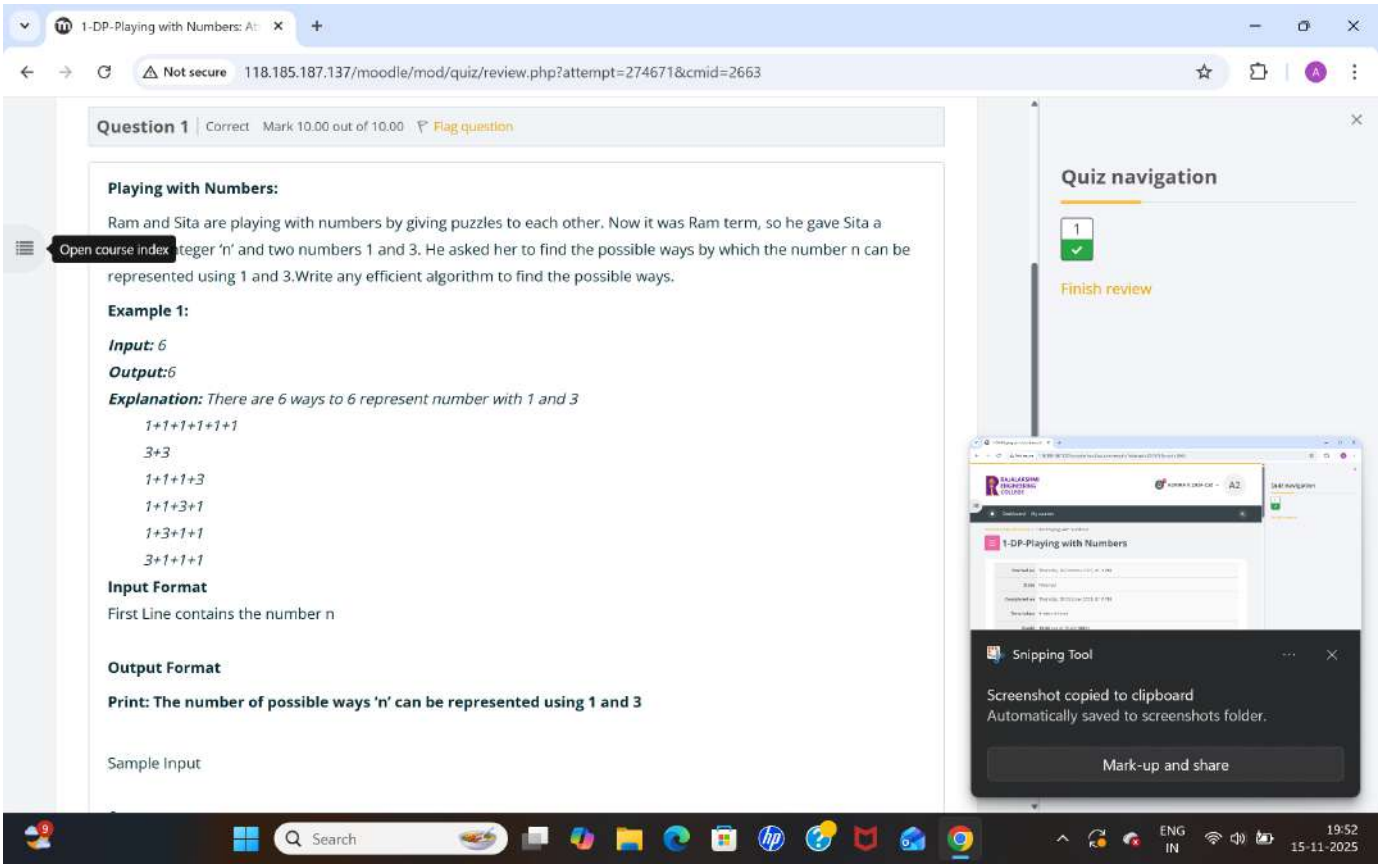



```c
int compareAsc(const void *a, const void *b) {
    return (*(int *)a - *(int *)b);
}


int compareDesc(const void *a, const void *b) {
    return (*(int *)b - *(int *)a);
}


int main() {
    int n;
    scanf("%d", &n);

    int array_One[n], array_Two[n];


    for (int i = 0; i < n; i++) {
        scanf("%d", &array_One[i]);
    }

    // Input array_Two
    for (int i = 0; i < n; i++) {
        scanf("%d", &array_Two[i]);
    }


    qsort(array_One, n, sizeof(int), compareAsc);


    qsort(array_Two, n, sizeof(int), compareDesc);


    long long minSum = 0;
    for (int i = 0; i < n; i++) {
        minSum += (long long)array_One[i] * array_Two[i];
    }

```


5-G-Product of Array elements
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233726&cmid=2662
}
printf("%lld\n", minSum);
return 0;
}
Open course index
Quiz navigation
1
Finish review
Input
Expected
Got
3 1 2 3 4 5 6
28
28
4 7 5 1 2 1 3
22
22
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG IN
19:51
15-11-2025


5-G-Product of Array elements
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=233726&cmid=2662
Open course index
3
4
1
5
590
590
30
10
40
8
9
4
3
10
Passed all tests!
Correct
Marks for this submission: 1.00/1.00.
Finish review
Back to Course
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:51
15-11-2025


1-DP-Playing with Numbers: A
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=274671&cmid=2663
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 1-DP-Playing with Numbers
1-DP-Playing with Numbers
Started on Thursday, 30 October 2025, 8:13 PM
State Finished
Completed on Thursday, 30 October 2025, 8:18 PM
Time taken 4 mins 50 secs
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 10.00 out of 10.00 Flag question
Playing with Numbers:
Quiz navigation
1
Finish review
Search
ENG IN
19:51 15-11-2025


1-DP-Playing with Numbers: A
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=274671&cmid=2663
Question 1
Correct
Mark 10.00 out of 10.00
Flag question
Playing with Numbers:
Ram and Sita are playing with numbers by giving puzzles to each other. Now it was Ram term, so he gave Sita a
teger 'n' and two numbers 1 and 3. He asked her to find the possible ways by which the number n can be
represented using 1 and 3.Write any efficient algorithm to find the possible ways.
Open course index
Example 1:
Input: 6
Output:6
Explanation: There are 6 ways to 6 represent number with 1 and 3
1+1+1+1+1+1
3+3
1+1+1+3
1+1+3+1
1+3+1+1
3+1+1+1
Input Format
First Line contains the number n
Output Format
Print: The number of possible ways 'n' can be represented using 1 and 3
Sample Input
Quiz navigation
1
Finish review
1-DP-Playing with Numbers
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:52
15-11-2025

1-DP-Playing with Numbers: A

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=274671&cmid=2663

Sample Output

6

Open course index

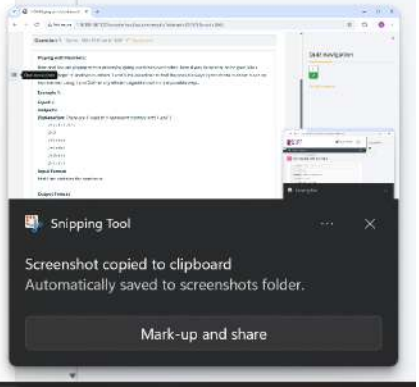
Answer: (penalty regime: 0 %)

```c
#include <stdio.h>

long long countWays(int n) {
    long long dp[n + 1];

    dp[0] = 1; // One way to make 0 (no numbers)

    for (int i = 1; i <= n; i++) {
        dp[i] = 0;
        if (i - 1 >= 0)
            dp[i] += dp[i - 1];
        if (i - 3 >= 0)
            dp[i] += dp[i - 3];
    }

    return dp[n];
}

int main() {
    int n;
    scanf("%d", &n);

    printf("%lld\n", countWays(n));

    return 0;
}
```

Quiz navigation

1

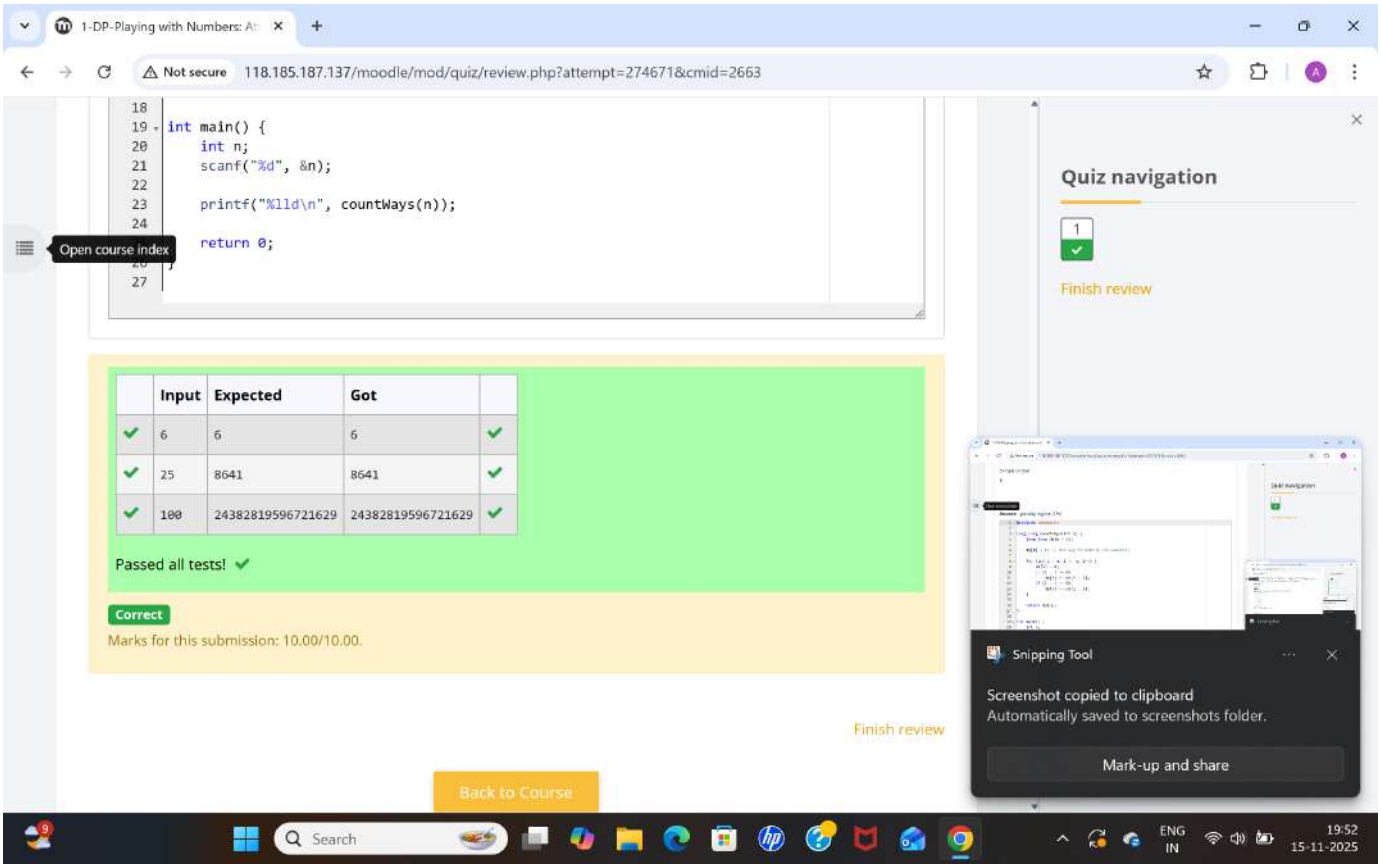
Finish review

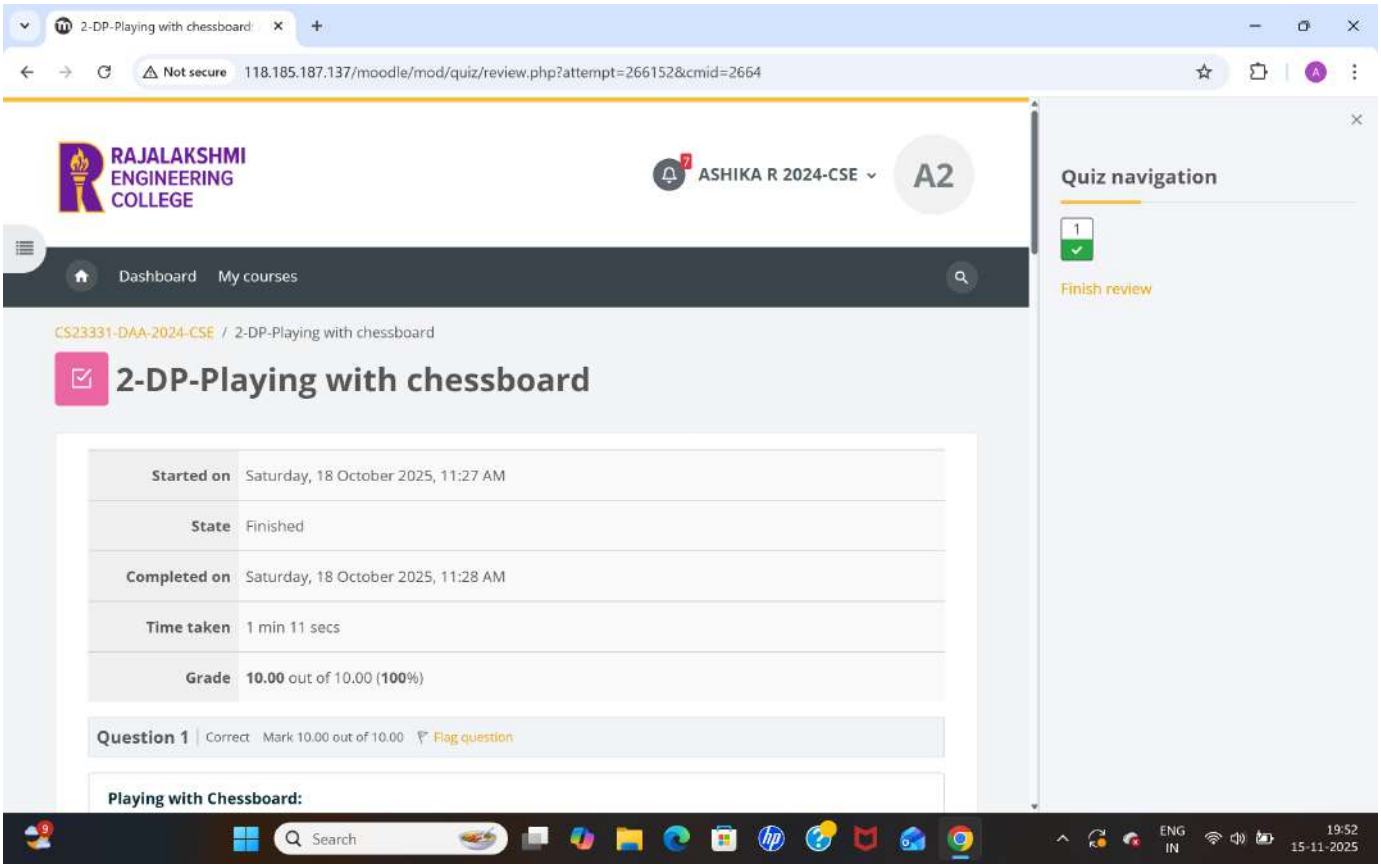


Search

ENG IN

19:52 15-11-2025


1-DP-Playing with Numbers: A
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=274671&cmid=2663
int main() {
int n;
scanf("%d", &n);
printf("%lld\n", countWays(n));
return 0;
Open course index
Quiz navigation
1
Finish review
Input
Expected
Got
6
6
6
25
8641
8641
100
24382819596721629
24382819596721629
Passed all tests!
Correct
Marks for this submission: 10.00/10.00.
Finish review
Back to Course
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:52
15-11-2025


2-DP-Playing with chessboard
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=266152&cmid=2664
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 2-DP-Playing with chessboard
2-DP-Playing with chessboard
Started on Saturday, 18 October 2025, 11:27 AM
State Finished
Completed on Saturday, 18 October 2025, 11:28 AM
Time taken 1 min 11 secs
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 10.00 out of 10.00 Flag question
Playing with Chessboard:
Quiz navigation
1
Finish review
Search
ENG IN
19:52
15-11-2025

2-DP-Playing with chessboard

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=266152&cmid=2664

Open course index

Playing with Chessboard:

Ram is given with an n*n chessboard with each cell with a monetary value. Ram stands at the (0,0), that the position of the top left white rook. He is been given a task to reach the bottom right black rook position (n-1, n-1) constrained eeds to reach the position by traveling the maximum monetary path under the condition that he can only travel one step right or one step down the board. Help ram to achieve it by providing an efficient DP algorithm.

Example:

Input

3

1 2 4

2 3 4

8 7 1

Output:

19

Explanation:

Totally there will be 6 paths among that the optimal is

Optimal path value:1+2+8+7+1=19

Input Format

First Line contains the integer n

The next n lines contain the n*n chessboard values

Output Format

Print Maximum monetary value of the path

Quiz navigation

1

Finish review

2-DP-Playing with chessboard

Snipping Tool

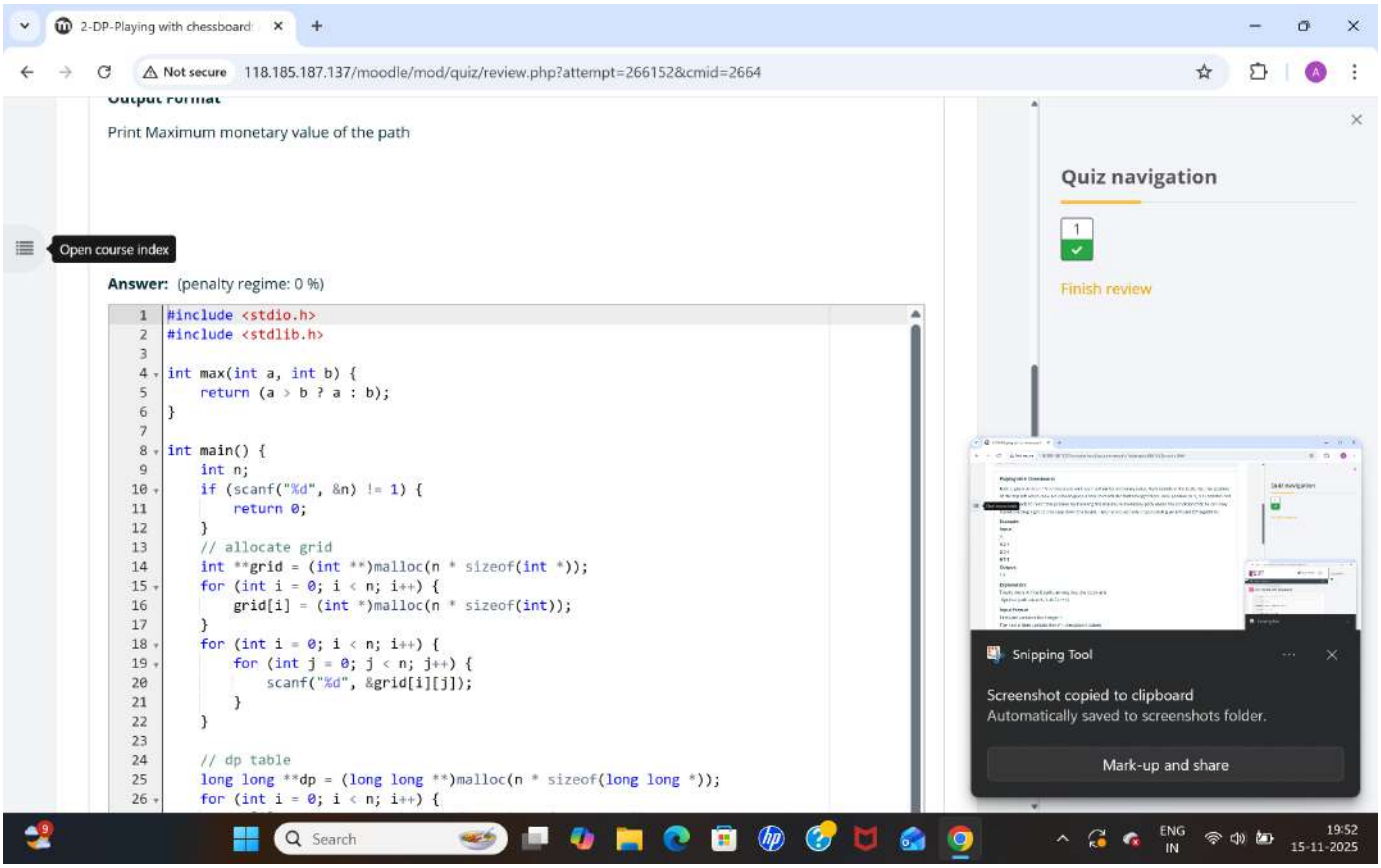
Screenshot copied to clipboard
Automatically saved to screenshots folder.

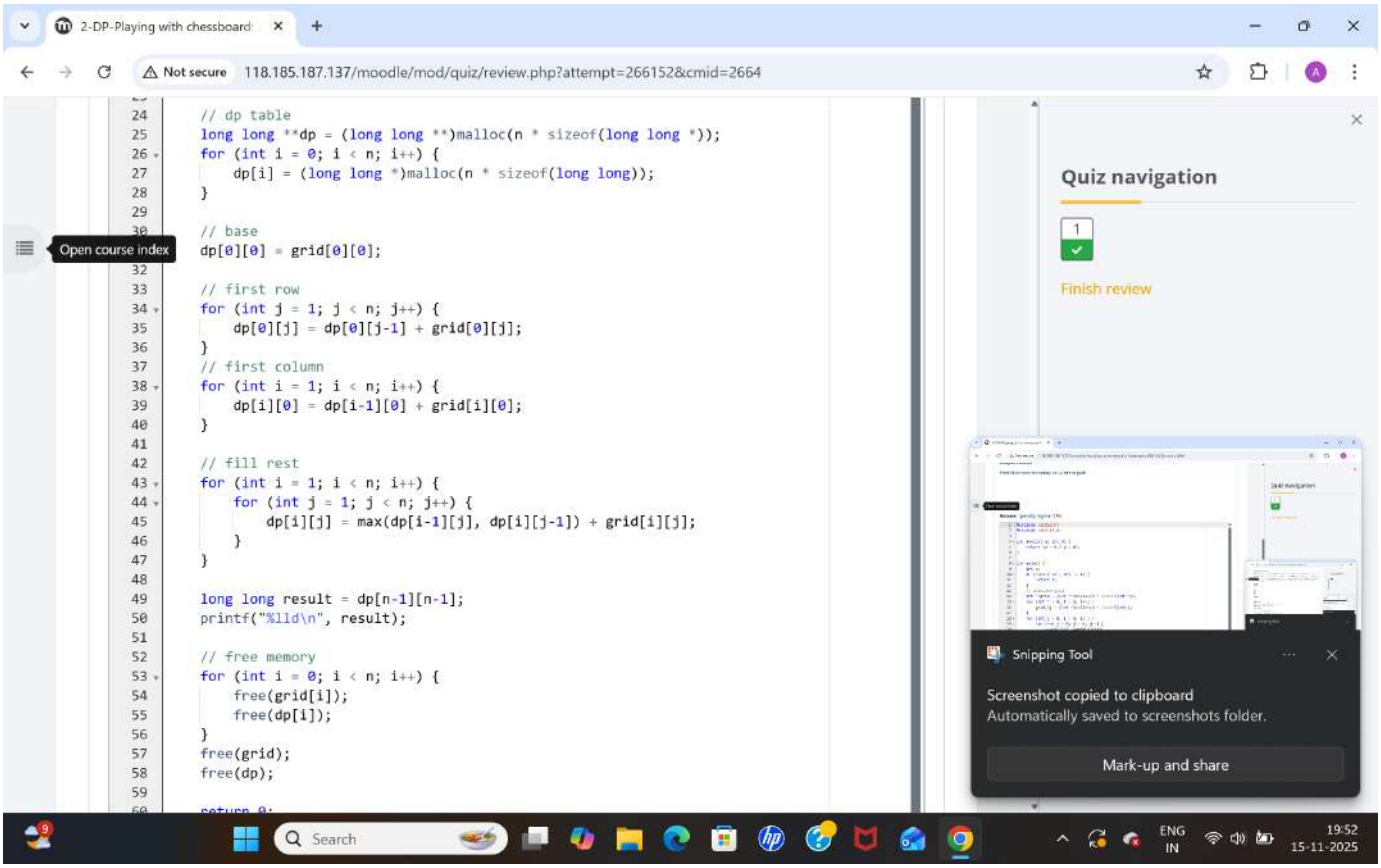
Mark-up and share

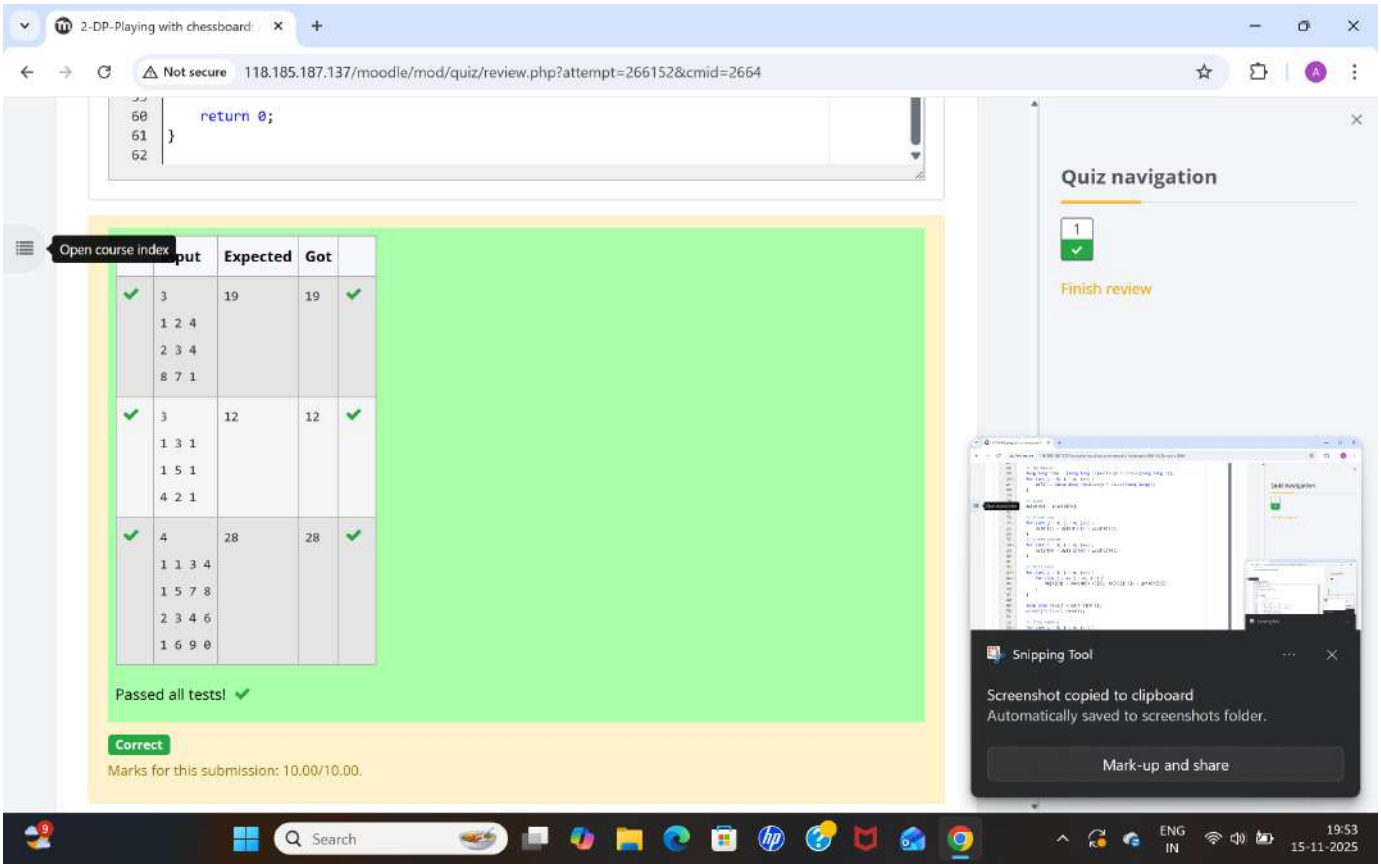
Search

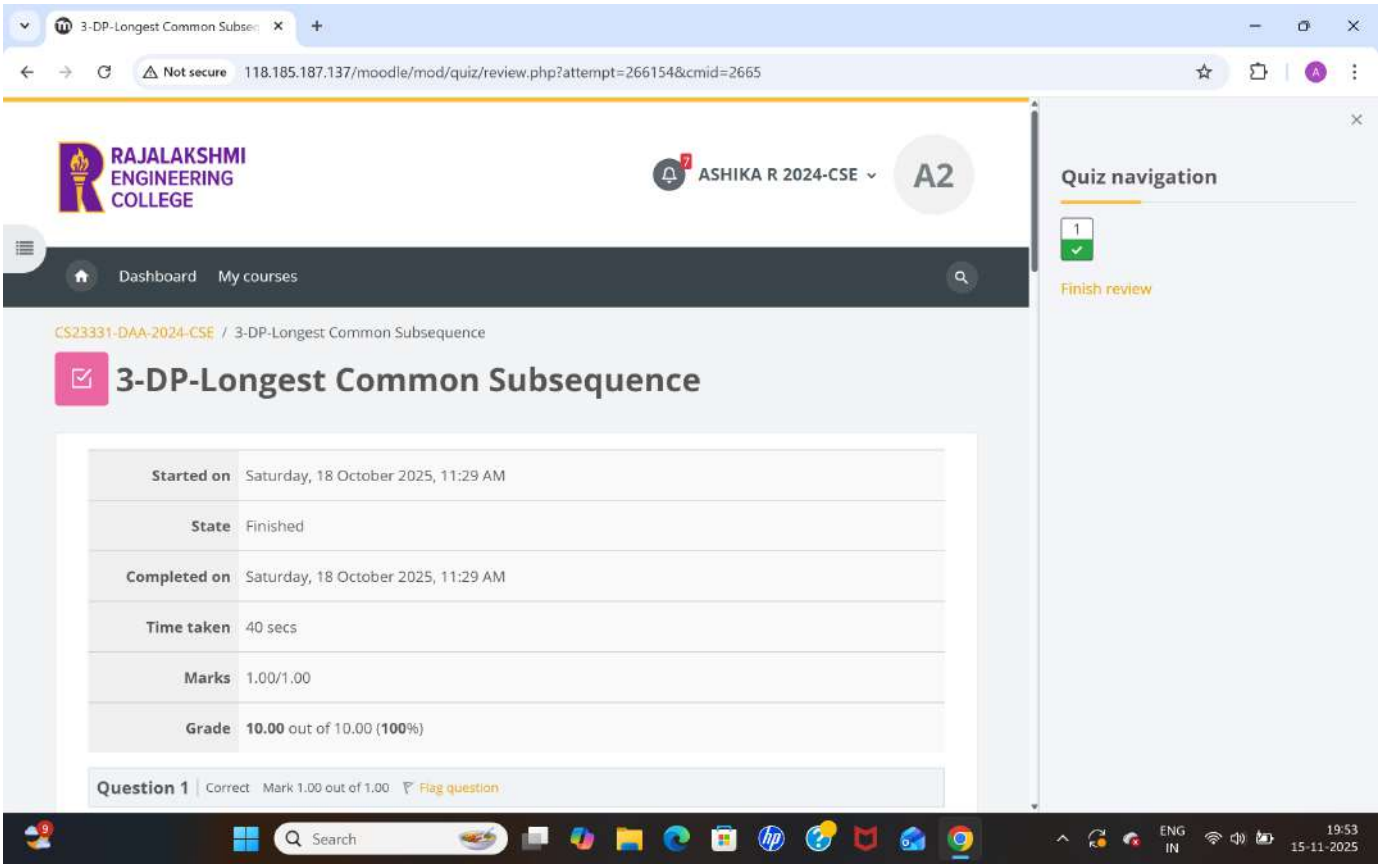
ENG
IN

19:52
15-11-2025


2-DP-Playing with chessboard
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=266152&cmid=2664
Print Maximum monetary value of the path
Open course index
Answer: (penalty regime: 0 %)
#include <stdio.h>
#include <stdlib.h>
int max(int a, int b) {
return (a > b ? a : b);
}
int main() {
int n;
if (scanf("%d", &n) != 1) {
return 0;
}
// allocate grid
int **grid = (int **)malloc(n * sizeof(int *));
for (int i = 0; i < n; i++) {
grid[i] = (int *)malloc(n * sizeof(int));
}
for (int i = 0; i < n; i++) {
for (int j = 0; j < n; j++) {
scanf("%d", &grid[i][j]);
}
}
// dp table
long long **dp = (long long **)malloc(n * sizeof(long long *));
for (int i = 0; i < n; i++) {
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:52
15-11-2025


2-DP-Playing with chessboard
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=266152&cmid=2664
Open course index
// dp table
long long **dp = (long long **)malloc(n * sizeof(long long *));
for (int i = 0; i < n; i++) {
dp[i] = (long long *)malloc(n * sizeof(long long));
}
// base
dp[0][0] = grid[0][0];
// first row
for (int j = 1; j < n; j++) {
dp[0][j] = dp[0][j-1] + grid[0][j];
}
// first column
for (int i = 1; i < n; i++) {
dp[i][0] = dp[i-1][0] + grid[i][0];
}
// fill rest
for (int i = 1; i < n; i++) {
for (int j = 1; j < n; j++) {
dp[i][j] = max(dp[i-1][j], dp[i][j-1]) + grid[i][j];
}
}
long long result = dp[n-1][n-1];
printf("%lld\n", result);
// free memory
for (int i = 0; i < n; i++) {
free(grid[i]);
free(dp[i]);
}
free(grid);
free(dp);
return 0;
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:52
15-11-2025


2-DP-Playing with chessboard
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=266152&cmid=2664
return 0;
}
Open course index
Expected
Got
19
19
12
12
28
28
Passed all tests!
Correct
Marks for this submission: 10.00/10.00.
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:53
15-11-2025


3-DP-Longest Common Subsec
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=266154&cmid=2665
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 3-DP-Longest Common Subsequence
3-DP-Longest Common Subsequence
Started on Saturday, 18 October 2025, 11:29 AM
State Finished
Completed on Saturday, 18 October 2025, 11:29 AM
Time taken 40 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
Search
ENG IN
19:53
15-11-2025

3-DP-Longest Common Subsec

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=266154&cmid=2665

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

Given two strings find the length of the common longest subsequence(need not be contiguous) between the two.

Open course index

s1: ggtabe

s2: tgatasb

s1	a	g	g	t	a	b	
s2	g	x	t	x	a	y	b

The length is 4

Solveing it using Dynamic Programming

For example:

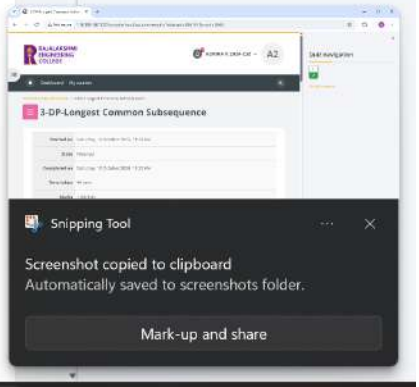
Input	Result
aab azb	2

Answer: (penalty regime: 0 %)

Quiz navigation

1

Finish review



Search

ENG IN 19:53 15-11-2025

3-DP-Longest Common Subse

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=266154&cmid=2665

Answer: (penalty regime: 0 %)

Open course index

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int max(int a, int b) {
    return (a > b ? a : b);
}

int main() {
    char s1[1001], s2[1001];
    // Read two strings (space separated or newline separated)
    if (scanf("%1000s %1000s", s1, s2) != 2) {
        return 0;
    }
    int m = strlen(s1);
    int n = strlen(s2);
    // Create DP table of size (m+1) x (n+1)
    int **dp = (int **)malloc((m+1) * sizeof(int *));
    for (int i = 0; i <= m; i++) {
        dp[i] = (int *)malloc((n+1) * sizeof(int));
    }
    // Initialize first row/col to 0
    for (int i = 0; i <= m; i++) {
        dp[i][0] = 0;
    }
    for (int j = 0; j <= n; j++) {
        dp[0][j] = 0;
    }
    // Fill dp table
    for (int i = 1; i <= m; i++) {
        for (int j = 1; j <= n; j++) {
            if (s1[i-1] == s2[j-1]) {
                dp[i][j] = 1 + dp[i-1][j-1];
            } else {
```

Quiz navigation

1

Finish review

Snipping Tool

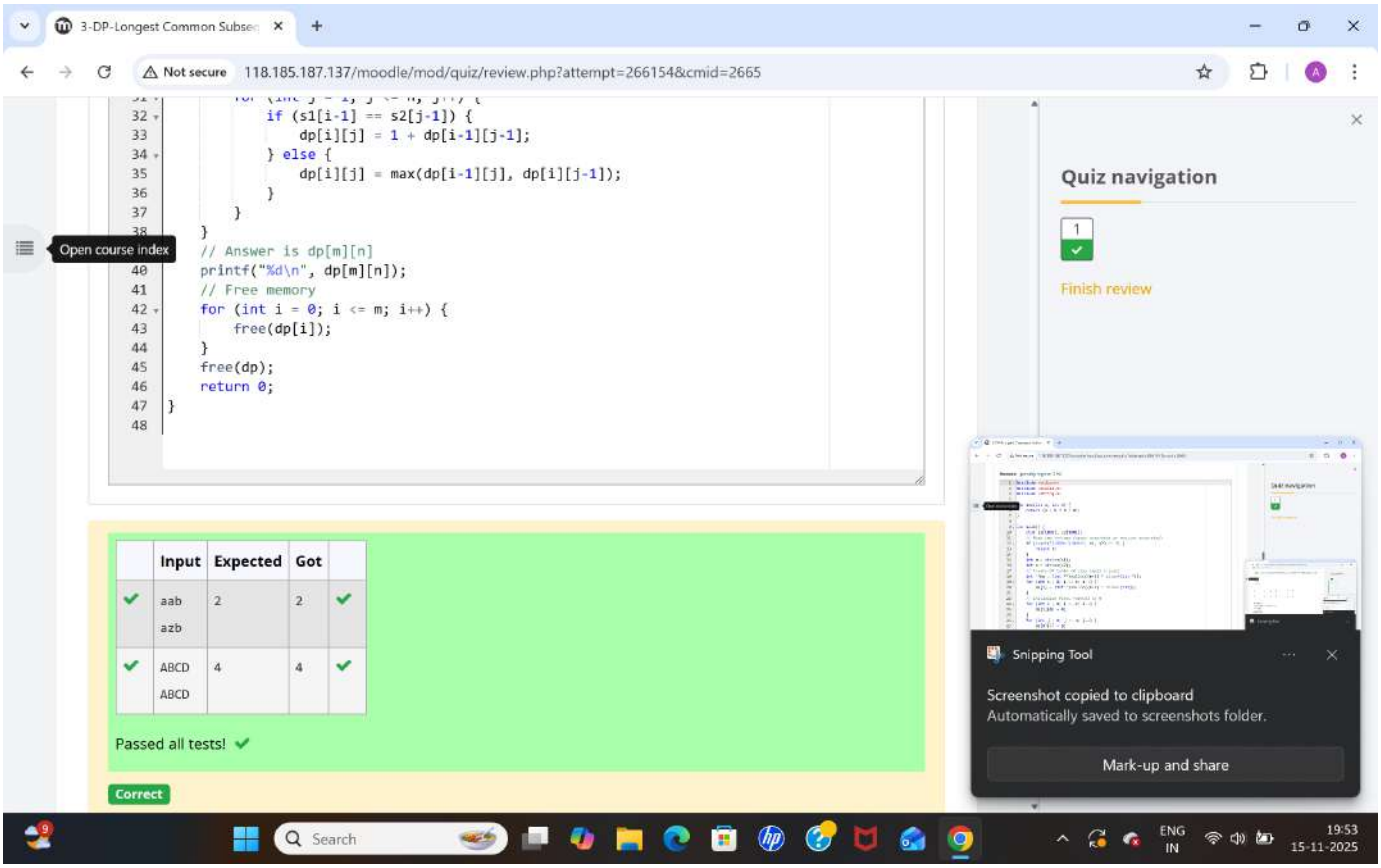
Screenshot copied to clipboard
Automatically saved to screenshots folder.

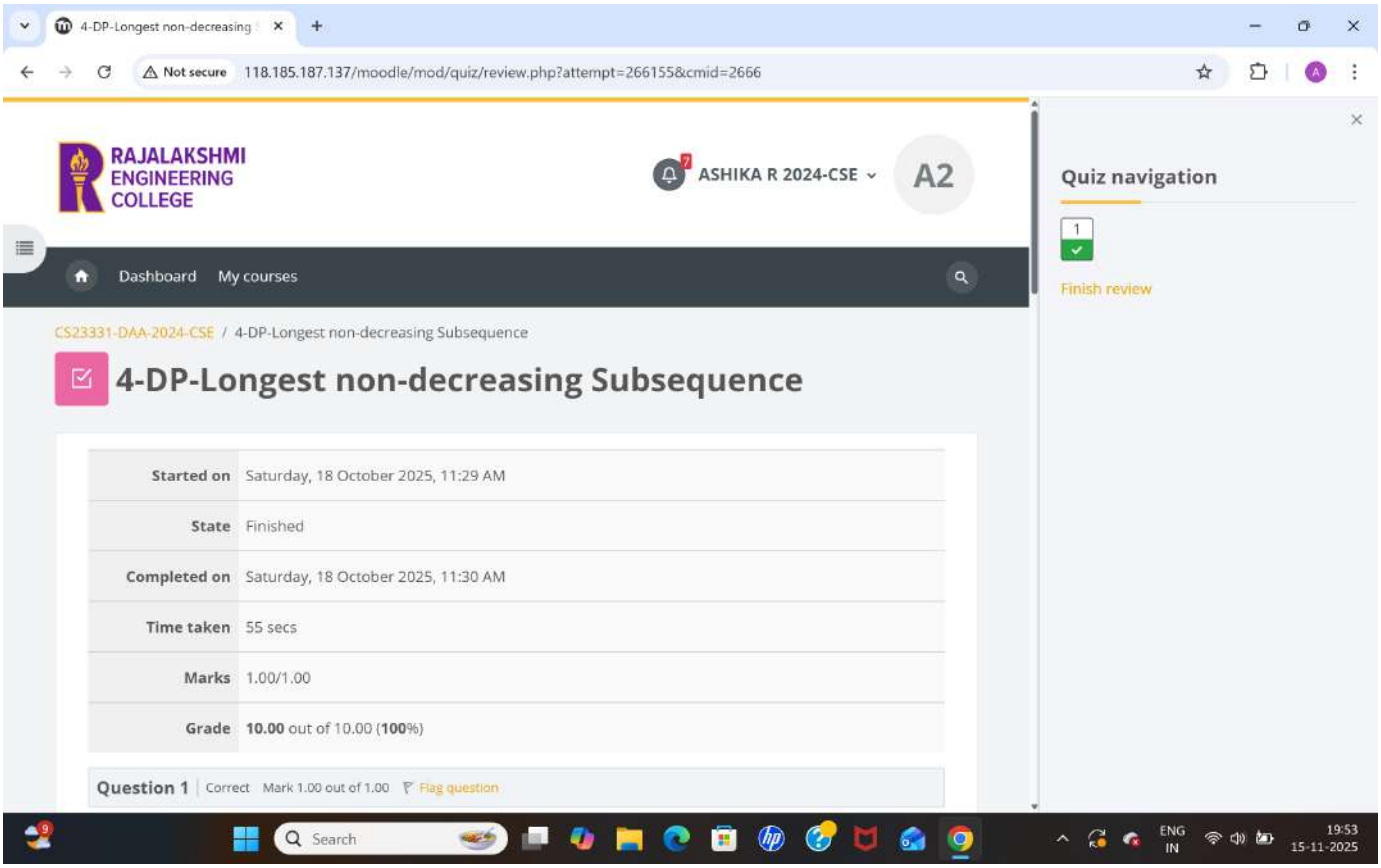
Mark-up and share

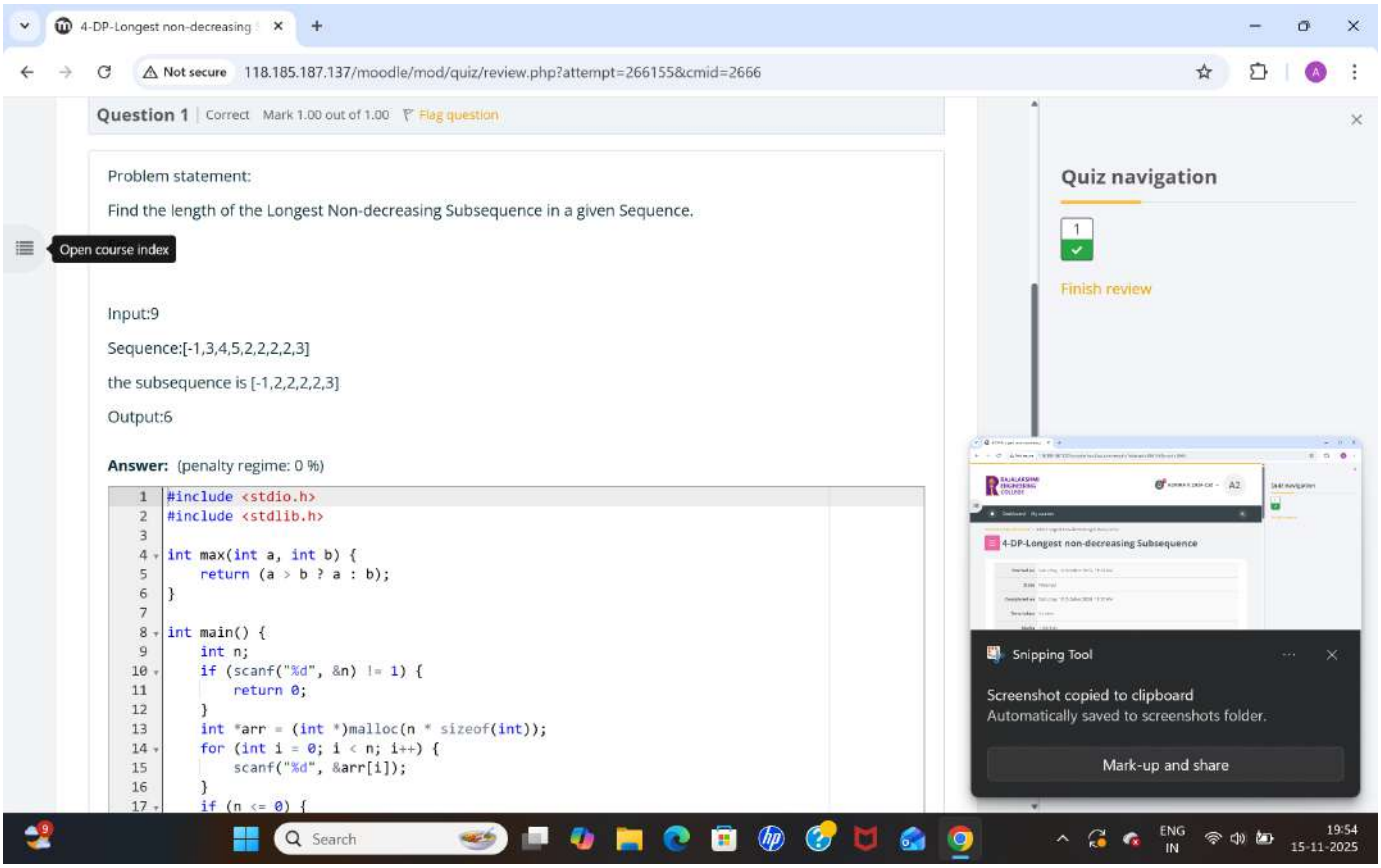
Search

ENG
IN

19:53
15-11-2025


3-DP-Longest Common Subsec
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=266154&cmid=2665
Open course index
Quiz navigation
1
Finish review
Input
Expected
Got
aab
azb
2
2
ABCD
ABCD
4
4
Passed all tests!
Correct
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:53
15-11-2025


4-DP-Longest non-decreasing
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=266155&cmid=2666
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 4-DP-Longest non-decreasing Subsequence
4-DP-Longest non-decreasing Subsequence
Started on Saturday, 18 October 2025, 11:29 AM
State Finished
Completed on Saturday, 18 October 2025, 11:30 AM
Time taken 55 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)
Question 1 Correct Mark 1.00 out of 1.00 Flag question
Quiz navigation
1
Finish review
Search
ENG IN
19:53
15-11-2025


4-DP-Longest non-decreasing
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=266155&cmid=2666
Question 1
Correct
Mark 1.00 out of 1.00
Flag question
Problem statement:
Find the length of the Longest Non-decreasing Subsequence in a given Sequence.
Open course index
Input:9
Sequence:[-1,3,4,5,2,2,2,2,3]
the subsequence is [-1,2,2,2,2,3]
Output:6
Answer: (penalty regime: 0 %)
#include <stdio.h>
#include <stdlib.h>
int max(int a, int b) {
return (a > b ? a : b);
}
int main() {
int n;
if (scanf("%d", &n) != 1) {
return 0;
}
int *arr = (int *)malloc(n * sizeof(int));
for (int i = 0; i < n; i++) {
scanf("%d", &arr[i]);
}
if (n <= 0) {
Quiz navigation
1
Finish review
4-DP-Longest non-decreasing Subsequence
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:54
15-11-2025

4-DP-Longest non-decreasing

Not secure 118.185.187.137/moodle/mod/quiz/review.php?attempt=266155&cmid=2666

Open course index

```c
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }
    if (n <= 0) {
        printf("0\n");
        free(arr);
        return 0;
    }

    int *dp = (int *)malloc(n * sizeof(int));
    for (int i = 0; i < n; i++) {
        dp[i] = 1;
    }

    int result = 1;
    for (int i = 1; i < n; i++) {
        for (int j = 0; j < i; j++) {
            if (arr[j] <= arr[i]) {
                dp[i] = max(dp[i], dp[j] + 1);
            }
        }
        if (dp[i] > result) {
            result = dp[i];
        }
    }

    printf("%d\n", result);

    free(arr);
    free(dp);
    return 0;
}
```

Quiz navigation

1

Finish review

Snipping Tool

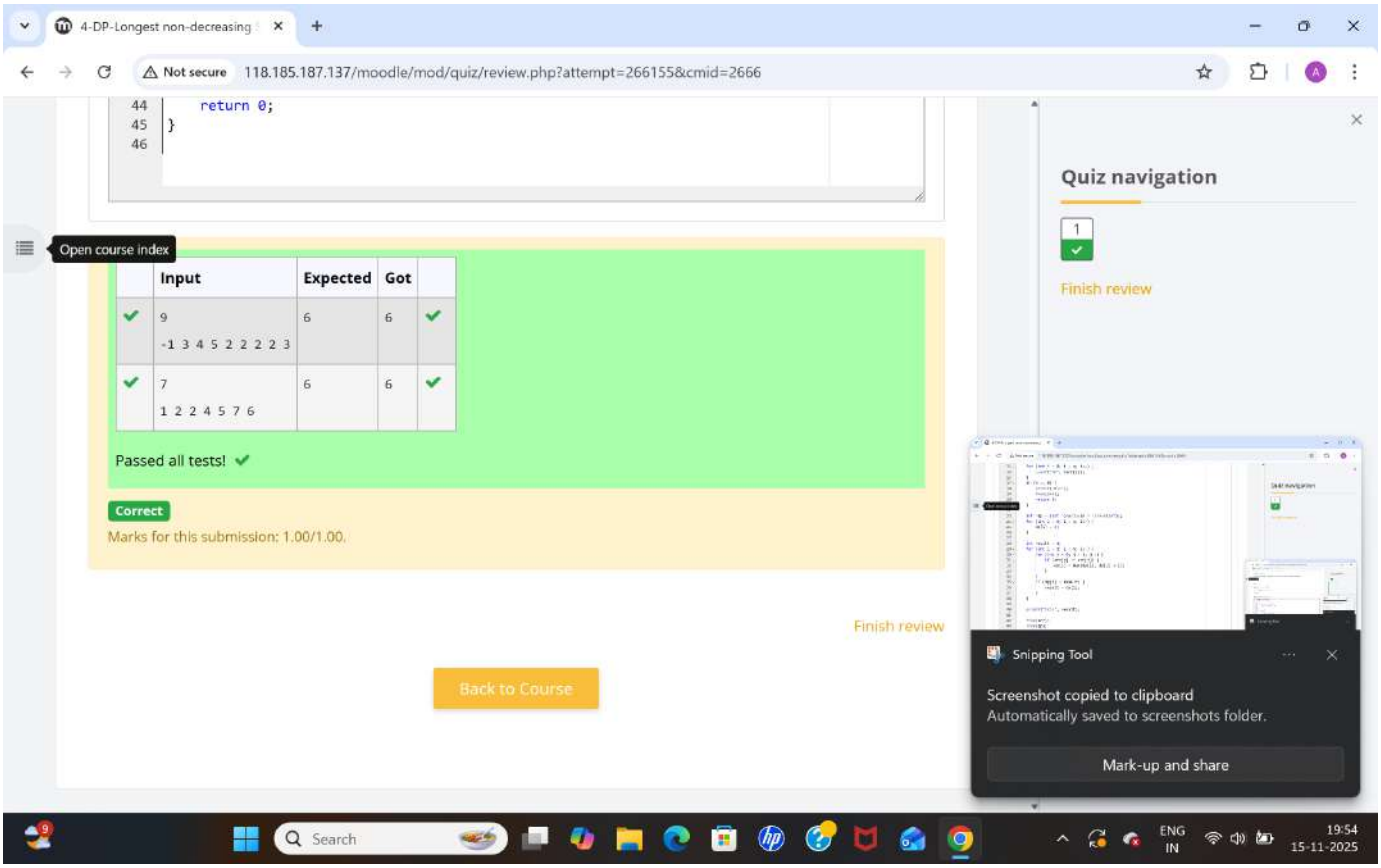
Screenshot copied to clipboard
Automatically saved to screenshots folder.

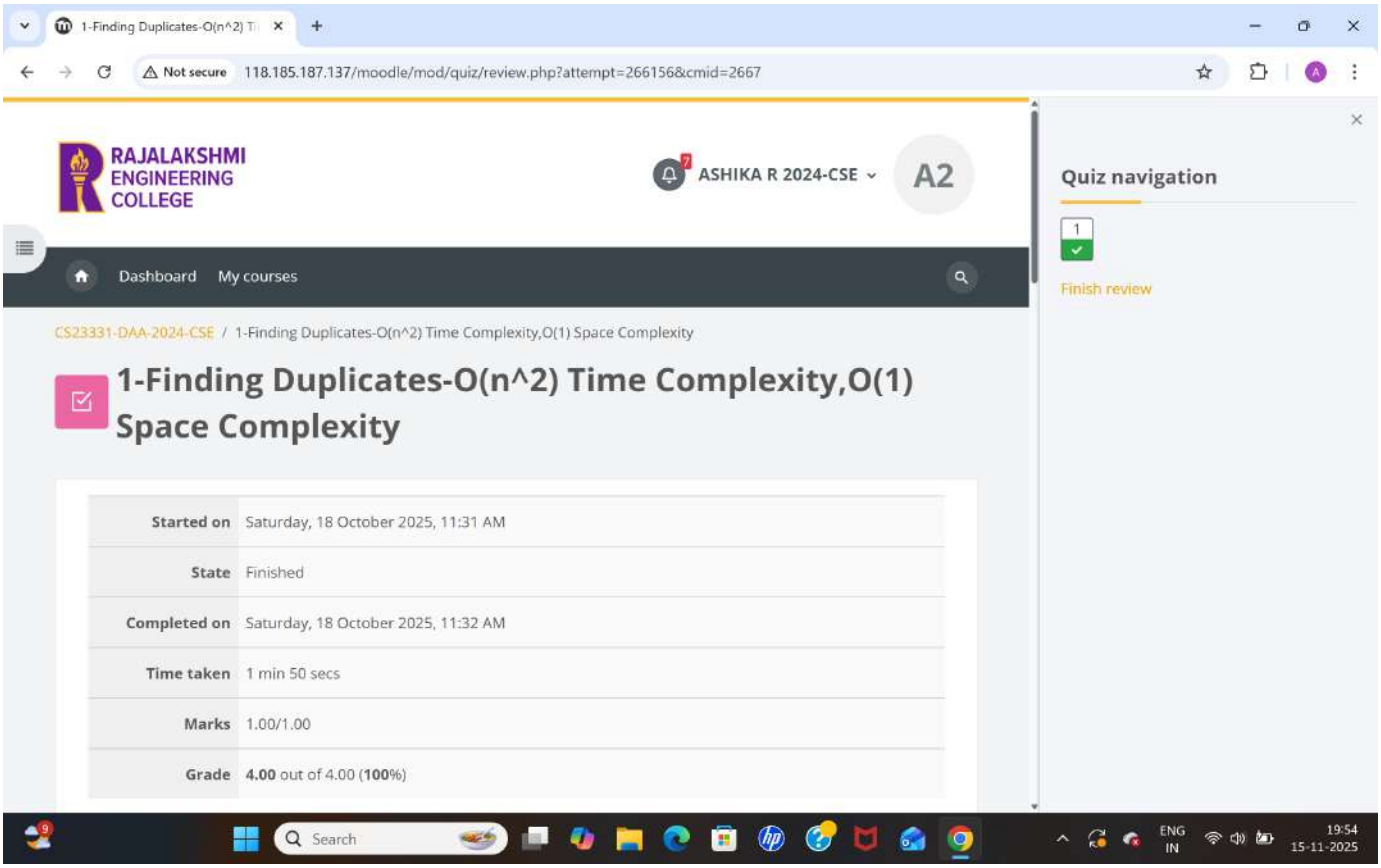
Mark-up and share

Search

ENG
IN

19:54
15-11-2025


4-DP-Longest non-decreasing
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=266155&cmid=2666
44 return 0;
45 }
46
Open course index
Input
Expected
Got
9
-1 3 4 5 2 2 2 2 3
6
6
7
1 2 2 4 5 7 6
6
6
Passed all tests!
Correct
Marks for this submission: 1.00/1.00.
Finish review
Back to Course
Quiz navigation
1
Finish review
Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Mark-up and share
Search
ENG
IN
19:54
15-11-2025


1-Finding Duplicates-O(n^2) Ti
Not secure
118.185.187.137/moodle/mod/quiz/review.php?attempt=266156&cmid=2667
RAJALAKSHMI ENGINEERING COLLEGE
ASHIKA R 2024-CSE
A2
Dashboard
My courses
CS23331-DAA-2024-CSE / 1-Finding Duplicates-O(n^2) Time Complexity,O(1) Space Complexity
1-Finding Duplicates-O(n^2) Time Complexity,O(1) Space Complexity
Started on Saturday, 18 October 2025, 11:31 AM
State Finished
Completed on Saturday, 18 October 2025, 11:32 AM
Time taken 1 min 50 secs
Marks 1.00/1.00
Grade 4.00 out of 4.00 (100%)
Quiz navigation
1
Finish review
Search
ENG
IN
19:54
15-11-2025