Week 15

Question 1:

Given an array of integers, reverse the given array in place using an index and loop rather than a built-in function.

Example

arr = [1, 3, 2, 4, 5]

Return the array [5, 4, 2, 3, 1] which is the reverse of the input array.

Function Description

Complete the function reverseArray in the editor below.

reverseArray has the following parameter(s):

int arr[n]: an array of integers

Return

int[n]: the array in reverse order

Constraints

1 ≤ n ≤ 100

0< arr[i] < 100

Input Format For Custom TestingProgram:

The first line contains an integer, n, the number of elements in arr.

Each line i of the n subsequent lines (where 0 < i < n) contains an integer, arr[i]

Sample Case 0

Sample Input For Custom Testing

5

1

3

2

4

5

Sample Output

5

4

3

2

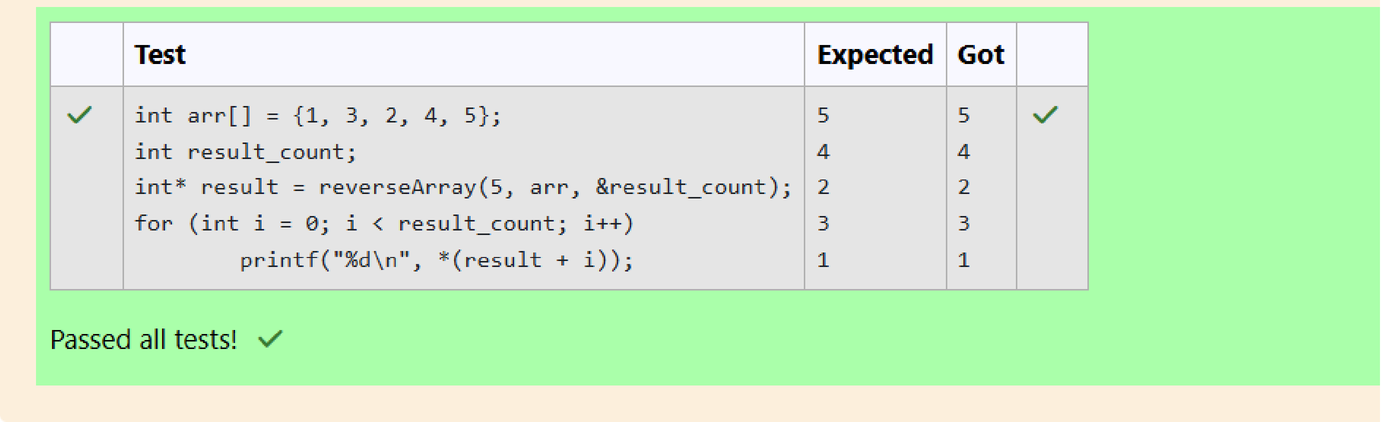
1

Status	Finished
Started	Monday, 13 January 2025, 7:26 PM
Completed	Monday, 13 January 2025, 7:41 PM
Duration	14 mins 35 secs

```
int* reverseArray(int arr_count, int *arr, int *result_count) {
*result_count=arr_count;
for(int i=0;i<arr_count/2;i++){
    int temp=arr[i];
    arr[i]=arr[arr_count-i-1];
    arr[arr_count-i-1]=temp;
}
return arr;

}
```

Output:



	Test	Expected	Got	

✓	`int arr[] = {1, 3, 2, 4, 5};`
	`int result_count;`
	`int* result = reverseArray(5, arr, &result_count);`
	`for (int i = 0; i < result_count; i++)`
	`        printf("%d\n", *(result + i));`
		5
		4
		2
		3
		1
			5
			4
			2
			3
			1
				✓

Passed all tests! ✓

Question 2:

Let’s print a chessboard!

Write a program that takes input: The first line contains T, the number of test cases

Each test case contains an integer N and also the starting character of the chessboard

Output Format

Print the chessboard as per the given examples

Sample Input 1:

2

2 W

3 B

Sample Output 1:

WB

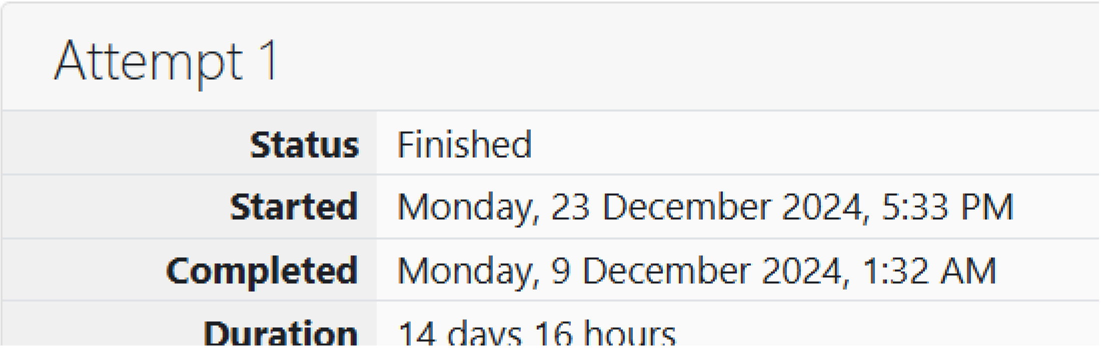
BW

BWB

WBW

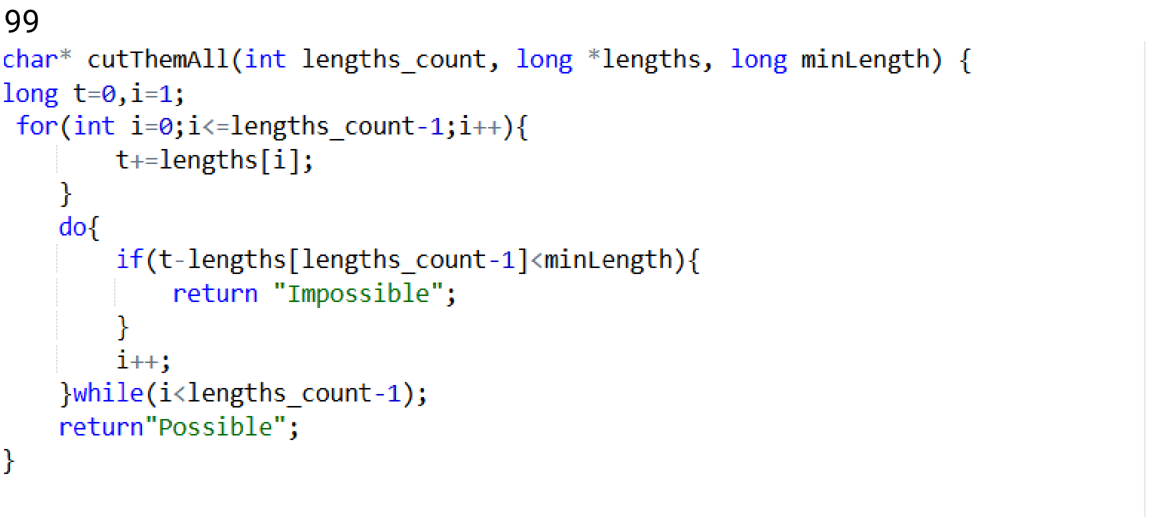
BWB

Program:



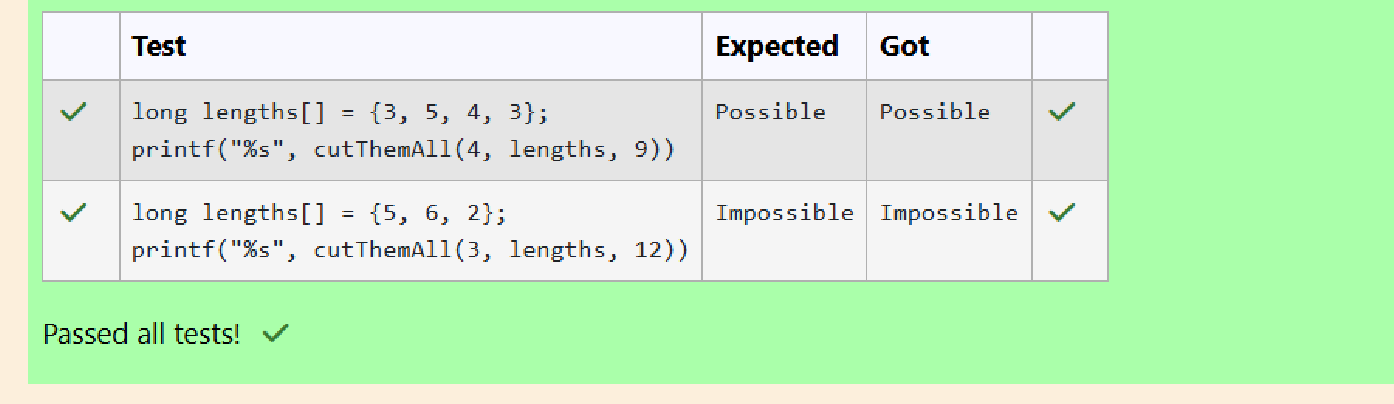
Attempt 1

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Monday, 9 December 2024, 1:32 AM
Duration	14 days 16 hours



```
99
char* cutThemAll(int lengths_count, long *lengths, long minLength) {
long t=0,i=1;
 for(int i=0;i<=lengths_count-1;i++){
        t+=lengths[i];
    }
    do{
        if(t-lengths[lengths_count-1]<minLength){
            return "Impossible";
        }
        i++;
    }while(i<lengths_count-1);
    return"Possible";
}
```

Output:



	Test	Expected	Got	

✓	`long lengths[] = {3, 5, 4, 3};`
	`printf("%s", cutThemAll(4, lengths, 9))`
		`Possible`	`Possible`	✓

✓	`long lengths[] = {5, 6, 2};`
	`printf("%s", cutThemAll(3, lengths, 12))`
		`Impossible`	`Impossible`	✓

Passed all tests! ✓