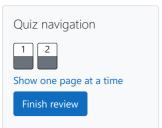
# GE23131-Programming Using C-2024



Status Finished
Started Monday, 13 January 2025, 10:52 AM
Completed Monday, 13 January 2025, 11:00 AM
Duration 8 mins 31 secs

Question **1**Correct
Marked out of 1.00

Flag

question

Given an array of integers, reverse the given array in place using an index and loop rather tha

#### 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

 $1 \le n \le 100$ 

 $0 < arr[i] \leq 100$ 

# **Input Format For Custom Testing**

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

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

# Sample Case 0

## Sample Input For Custom Testing

5

1

3

2

4

5

# Sample Output

5

4

2

3

1

#### Explanation

The input array is [1, 3, 2, 4, 5], so the reverse of the input array is [5, 4, 2, 3, 1].

#### Sample Case 1

#### Sample Input For Custom Testing

4

17

10

21

45

# Sample Output

45

21

10

17

Explanation

The input array is [17, 10, 21, 45], so the reverse of the input array is [45, 21, 10, 17].

Te	est	Expected	Got
ir	nt arr[] = {1, 3, 2, 4, 5};	5	5
ir	nt result_count;	4	4
ir	nt* result = reverseArray(5, arr, &result_count);	2	2
fo	or (int i = 0; i < result_count; i++)	3	3
	<pre>printf("%d\n", *(result + i));</pre>	1	1

Question **2**Correct
Marked out of 1.00

Flag question

An automated cutting machine is used to cut rods into segments. The cutting machine can o of *minLength* or more, and it can only make one cut at a time. Given the array *lengths[]* represof each segment, determine if it is possible to make the necessary cuts using this machine. The lengths already, in the order given.

# Example

n = 3 lengths = [4, 3, 2] minLength = 7

The rod is initially sum(lengths) = 4 + 3 + 2 = 9 units long. First cut off the segment of length 7 = 2. Then check that the length 7 rod can be cut into segments of lengths 4 and 3. Since 7 to minLength = 7, the final cut can be made. Return "Possible".

## Example

n = 3 lengths = [4, 2, 3] minLength = 7

The rod is initially sum(lengths) = 4 + 2 + 3 = 9 units long. In this case, the initial cut can be o Regardless of the length of the first cut, the remaining piece will be shorter than minLength. E cannot be made, the answer is "lmpossible".

## **Function Description**

Complete the function *cutThemAll* in the editor below.

cutThemAll has the following parameter(s):
int lengths[n]: the lengths of the segments, in order
int minLength: the minimum length the machine can accept

Returns

**REC-CIS** 

#### Constraints

- $\cdot \quad 2 \le n \le 10^5$
- $1 \le t \le 10^9$
- $1 \le lengths[i] \le 10^9$
- · The sum of the elements of lengths equals the uncut rod length.

#### **Input Format For Custom Testing**

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

Each line i of the n subsequent lines (where  $0 \le i < n$ ) contains an integer, lengths[i].

The next line contains an integer, minLength, the minimum length accepted by the machine.

#### Sample Case 0

#### **Sample Input For Custom Testing**

```
STDIN Function
-----
4 → lengths[] size n = 4
3 → lengths[] = [3, 5, 4, 3]
5
4
3
9 → minLength= 9
```

# **Sample Output**

Possible

#### **Explanation**

The uncut rod is 3 + 5 + 4 + 3 = 15 units long. Cut the rod into lengths of 3 + 5 + 4 = 12 and piece into lengths 3 and 5 + 4 = 9. The remaining segment is 5 + 4 = 9 units and that is long cut.

#### Sample Case 1

# **Sample Input For Custom Testing**

```
STDIN Function
-----
3 → lengths[] size n = 3
5 → lengths[] = [5, 6, 2]
6
2
12 → minLength= 12
```

## **Sample Output**

Impossible

# **Explanation**

REC-CIS