

Week-15-Pointers: Attempt review | REC-CIS - Google Chrome

Not secure rajalakshmicolleges.org/moodle/mod/quiz/review.php?attempt=164328&cmid=404

REC-CIS

```
22 * int* return_integer_array_using_dynamic_allocation(int* result_count) {
23 *     *result_count = 5;
24 *
25 *     int *a = malloc(5 * sizeof(int));
26 *
27 *     for (int i = 0; i < 5; i++) {
28 *         *(a + i) = i + 1;
29 *     }
30 *
31 *     return a;
32 * }
33 *
34 */
35 int* reverseArray(int arr_count, int *arr, int *result_count) {
36     *result_count=arr_count;
37     for(int i=0;i<arr_count/2;i++){
38         int temp=arr[i];
39         arr[i]=arr[arr_count-i-1];
40         arr[arr_count-i-1]=temp;
41     }
42     return arr;
43 }
44
```

Test	Expected	Got
✓ int arr[] = {1, 3, 2, 4, 5}; int result_count;	5 4	5 4 ✓

Constants Variables

Search

ENG IN 9:51 PM 1/15/2025

Week-15-Pointers: Attempt review | REC-CIS - Google Chrome

Not secure rajalakshmicolleges.org/moodle/mod/quiz/review.php?attempt=164328&cmid=404

REC-CIS

```
33 */
34
35 int* reverseArray(int arr_count, int *arr, int *result_count) {
36     *result_count=arr_count;
37     for(int i=0;i<arr_count/2;i++){
38         int temp=arr[i];
39         arr[i]=arr[arr_count-i-1];
40         arr[arr_count-i-1]=temp;
41     }
42     return arr;
43 }
44
```

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

An automated cutting machine is used to cut rods into segments. The cutting machine can only hold a rod of *minLength*

Constants Variables

Search

ENG IN 9:51 PM 1/15/2025

Week-15-Pointers: Attempt review | REC-CIS - Google Chrome

Not secure rajalakshmicolleges.org/moodle/mod/quiz/review.php?attempt=164328&cmid=404

REC-CIS

```
27
28 */
29 char* cutThemAll(int lengths_count, long *lengths, long minLength) {
30     long t=0,i=1;
31     for(int i=0;i<=lengths_count-1;i++){
32         t+=lengths[i];
33     }
34     do{
35         if(t-lengths[lengths_count-i-1]<minLength){
36             return "Impossible";
37         }
38         i++;
39     }while(i<lengths_count-1);
40     return "Possible";
41 }
42
```

	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! ✓

Constants Variables

Search

ENG IN 9:51 PM 1/15/2025