```
 *      - Allocate the array statically or
 *
 * For example,
 * int* return_integer_array_using_static
 *     *result_count = 5;
 *
 *     static int a[5] = {1, 2, 3, 4, 5};
 *
 *     return a;
 * }
 *
 * int* return_integer_array_using_dynami
 *     *result_count = 5;
 *
 *     int *a = malloc(5 * sizeof(int));
 *
 *     for (int i = 0; i < 5; i++) {
 *         *(a + i) = i + 1;
 *     }
 *
 *     return a;
 * }
 *
 */
int* reverseArray(int arr_count, int *arr
    *result_count=arr_count;
    int *reversed=(int*)malloc(arr_count*
    if(reversed==NULL)exit(1);
    for(int i=0;i<arr_count;i++)
    reversed[i]=arr[arr_count-1-i];

    return reversed;
}
```

	Expected	Got	
4, 5};	5	5	✓
	4	4	
Array(5, arr, &result_count);	2	2	
result_count; i++)	3	3	
, *(result + i));	1	1	

Passed all tests! ✓

```c
/*
 * To return the string from the function
 *
 * For example,
 * char* return_string_using_static_alloc
 *     static char s[] = "static allocati
 *
 *     return s;
 * }
 *
 */
char* cutThemAll(int lengths_count, long
    long totallength=0;
    for(int i=0;i<lengths_count;i++)
    totallength+=lengths[i];
    long currentlength=0;
    for(int i=0;i<lengths_count-1;i++){
        currentlength+=lengths[i];
        long rem_length=totallength-curre
        if(rem_length>minLength)return "P

    }return "Impossible";
}
```

	Expected	Got	
5, 4, 3}; All(4, lengths, 9))	Possible	Possible	✓
6, 2}; All(3, lengths, 12))	Impossible	Impossible	✓

Passed all tests! ✓