Array

October 3, 2024

Arrays

[1]:

```
# 1. Given an array, check if it contains any duplicates or not.
# arr = [1, 2, 4, 2, 5, 9]
# Output = True

def has_duplicates(arr):
    return len(arr) != len(set(arr))

arr = [1, 2, 4, 2, 5, 9]
output = has_duplicates(arr)
print(output)
```

```
True
```

[3]:

```
# 2. Given an array and an integer k, rotate the array to the right by k steps.
# arr = [1, 2, 3, 4, 5, 6, 7] k = 3
# Output = [5, 6, 7, 1, 2, 3, 4]
def rotate_array(arr, k):
    k = k % len(arr)
    return arr[-k:] + arr[:-k]

arr = [1, 2, 3, 4, 5, 6, 7]
k = 3
output = rotate_array(arr, k)
print(output)
```

```
[5, 6, 7, 1, 2, 3, 4]
```

[5]:

```
# 3. Reverse the given array in-place, means without using any extra data␣
  ↪structure.
# arr = [2, 4, 5, 7, 9, 12]
# Output = [12, 9, 7, 5, 4, 2]
def reverse_array(arr):
    left = 0
    right = len(arr) - 1

    while left < right:
        arr[left], arr[right] = arr[right], arr[left]
        left += 1
        right -= 1

arr = [2, 4, 5, 7, 9, 12]
reverse_array(arr)
print(arr)
```

```
[12, 9, 7, 5, 4, 2]
```

[7]:

```
# 4.  Given an array of integers, find the maximum element in an array
# arr = [10, 5, 20, 8, 15]
# Output = 20
arr = [10, 5, 20, 8, 15]
output = max(arr)
print(output)
```

```
20
```

[9]:

```
# 5. Given a sorted array, remove the duplicate element without using any extra␣
  ↪data structure.
# arr = [1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5]
# Output = [1, 2, 3, 4, 5]
def remove_duplicates(arr):
    if not arr:
        return 0

    write_index = 1

    for i in range(1, len(arr)):
        if arr[i] != arr[i - 1]:
            arr[write_index] = arr[i]
            write_index += 1

    return arr[:write_index]

arr = [1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 5, 5]
output = remove_duplicates(arr)
print(output)
```

```
[1, 2, 3, 4, 5]
```

[ ]:

```
#
```

[ ]:

```
#
```

[ ]:

```
#
```

[ ]:

```
#
```

[ ]:

```
#
```

[ ]:

```
#
```

[ ]:

```
#
```

[ ]:

```
#
```