Problem Set 3 Exercise #08: Rotate Right K

Reference: Lecture 7 notes

Learning objective: One-dimensional array

Estimated completion time: 30 minutes

Problem statement:

[CS1010 AY2012/13 Semester 1 Midterm Test, Q9]

Write a program **rotate_right.c** that contain a function

```
    void rotate_right_k(int arr[], int size, int k)
```

that takes an array **`arr`** of **`size`** (1 ≤ **`size`** ≤ 100) elements, rotate it **`k`** positions to the right, where **`k`** is a positive integer. For example, given an array **`arr`** {1, 2, 3, 4, 5, 6},

- Calling **`rotate_right_k(arr, 6, 1)`** would result in **`arr`** becoming {6, 1, 2, 3, 4, 5}.
- Calling **`rotate_right_k(arr, 6, 3)`** would result in **`arr`** becoming {4, 5, 6, 1, 2, 3}.
- Calling **`rotate_right_k(arr, 6, 60002)`** would result in **`arr`** becoming {5, 6, 1, 2, 3, 4}.

A tip is given at the beginning of next page.

Sample run #1:

```
Enter the number of elements: 6
Enter 6 elements: 1 2 3 4 5 6
Enter k: 1
[6, 1, 2, 3, 4, 5]
```

Sample run #2:

```
Enter the number of elements: 6
Enter 6 elements: 1 2 3 4 5 6
Enter k: 3
[4, 5, 6, 1, 2, 3]
```

Sample run #3:

```
Enter the number of elements: 6
Enter 6 elements: 1 2 3 4 5 6
Enter k: 60002
5 6 1 2 3 4
```

Useful tip:

It might be useful to write another function

```
    void rotate_right_1(int arr[], int size)
```

that rotates the array **`arr`** to the right by 1 position.

It may be called by the **`rotate_right_k()`** function.