

# CS 284: Homework Assignment 6 (Optional)

Due: May 5th, 11:59pm

## 1 Assignment Policies

**Collaboration Policy.** Homework will be done individually: each student must hand in their own answers. It is acceptable for students to collaborate in understanding the material but not in solving the problems or programming. Use of the Internet is allowed, but should not include searching for existing solutions.

**Under absolutely no circumstances code can be exchanged between students.** Excerpts of code presented in class can be used.

**Assignments from previous offerings of the course must not be re-used.** Violations will be penalized appropriately.

**Late Policy.** No late submissions will be allowed.

**This assignment will replace one of your lowest homework grades**

## 2 Assignment

In this assignment you are asked to implement *Counting Sort*. This algorithm takes as input an array of  $n$  integers in the range of 0 to  $k$ . Its running time is  $\mathcal{O}(n)$ , and hence is more efficient than all algorithms seen in class. In contrast to those seen in class, *Counting Sort* does not use comparisons.

The signature of the main class should be `public class CountingSort` and is supposed to contain the following method:

```
public static void sort(int[] A)
```

Next we present the pseudocode for `COUNTING_SORT (A, B, k)`, taken from [?]. `A` is the array `A[1..n]` to be sorted and has length  $n$ . Also, its elements are integers in the range  $0..k$ . `B` is an array `B[1..n]` that will hold the sorted output. `c` is an auxiliary array `c[0..k]` for temporary storage. You should use this as the basis for your Java implementation. Once you have implemented it, include some test cases.

```

10 for i = 0 to k do
2   C[i] = 0
   for j = 1 to n do
4     C[A[j]] = C[A[j]] + 1
     //C[i] now contains the number of elements equal to i
6   for i = 2 to k do
     C[i] = C[i] + C[i-1]
8   // C[i] now contains the number of elements <= i
   for j = n downto 1 do
10    B[C[A[j]]] = A[j]
    C[A[j]] = C[A[j]] - 1

```

Here is an example of a run. The array to be sorted is A whose size is  $n = 8$ .

A	2	5	3	0	2	3	0	3
---	---	---	---	---	---	---	---	---

The integers in A are in the range 0 to  $k = 5$ . Here is a trace:

- At line 5:

C	2	0	2	3	0	1
---	---	---	---	---	---	---

- At line 8:

C	2	2	4	7	7	8
---	---	---	---	---	---	---

- Iteration 1 of loop in line 9:

B							3		C	2	2	4	6	7	8
---	--	--	--	--	--	--	---	--	---	---	---	---	---	---	---

- Iteration 2 of loop in line 9:

B		0					3		C	1	2	4	6	7	8
---	--	---	--	--	--	--	---	--	---	---	---	---	---	---	---

- Iteration 3 of loop in line 9:

B		0				3	3		C	1	2	4	5	7	8
---	--	---	--	--	--	---	---	--	---	---	---	---	---	---	---

- After all iterations of loop in line 9:

B	0	0	2	2	3	3	3	5
---	---	---	---	---	---	---	---	---

### 3 Submission instructions

Please archive your CountingSort.java and the CountingSortTest.java files in a single ZIP file and submit your ZIP file on Canvas. No report is required. Some further guidelines:

- Use JavaDoc to comment your code.
- Check the arguments of methods.