Exercise Session 05

Exercise 1.
Consider the MAX-HEAPIFY procedure (CLRS, pp. 154) as defined below.

```
MAX-HEAPIFY(A, i)
l = LEFT(i)
r = RIGHT(i)
if l ≤ A.heap-size and A[l] > A[i]
    largest = l
else
    largest = i
if r ≤ A.heap-size and A[r] > A[largest]
    largest = r
if largest ≠ i
    exchange A[i] with A[largest]
    MAX-HEAPIFY(A, largest)
```

Use induction to prove that MAX-HEAPIFY is correct.

Exercise 2.
Starting with the procedure MAX-HEAPIFY (CLRS, pp. 154), write pseudocode for the procedure MIN-HEAPIFY(A, i), which performs the corresponding manipulation on a min-heap. How does the running time of MIN-HEAPIFY compare to that of MAX-HEAPIFY?

Exercise 3.
Consider the pseudocode of the PARTITION procedure.

```
PARTITION(A, p, r)
x = A[r]
i = p - 1
for j = p to r - 1
    if A[j] ≤ x
        i = i + 1
        exchange A[i] with A[j]
exchange A[i + 1] with A[r]
return i + 1
```

Assume that all elements in the array $A[p\,..\,r]$ are equal, that is, $A[p] = A[p+1] = \cdots = A[r]$. What value will PARTITION(A, p, r) return? How does QUICKSORT perform on arrays that have the same value compared with INSERTION-SORT and MERGESORT?

Exercise 4.
Modify the pseudocode of the PARTITION procedure so that the QUICKSORT algorithm (CLRS, pp. 171) will sort in nonincreasing order. Argument about the correctness of your solution.

Exercise 5.
Consider the pseudocode of COUNTING-SORT (CLRS, pp. 195)

Counting-Sort(A, B, k)

```
1   let C[0..k] be a new array
2   for j − 1 to k
3       C[i] = 0
4   for j = 1 to A.length
5       C[A[j]] = C[A[j]] + 1
6   for i = 1 to k
7       C[i] = C[i] + C[i − 1]
8   for j = A.length downto 1
9       B[C[A[j]]] = A[j]
10      C[A[j]] = C[A[j]] − 1
```

Modify the above pseudocode by replacing the **for**-loop header in line 8 as

```
8   for j = 1 to A.length
```

Is the modified algorithm stable?

★ **Exercise 6.**

Use induction to prove that Radix-Sort is correct. Where does your proof need the assumption that the intermediate sorting procedure is stable? Justify your answer.

Exercise 7.

Assume to use Quicksort as the sorting subroutine for Radix-Sort. Will the resulting procedure be correct? Justify your answer.