Exercise Sheet 6

Handout: Oct 14th — Deadline: Oct 23rd - 4pm

Question 6.1 (marks 0.5)

Consider the following algorithm. Does it sort correctly? (You might want to work out your own example to understand this better.)

DO-I-SORT(A, n)

```
1: for i = 1 to n do
2:     for j = 1 to n do
3:         if A[i] < A[j] then
4:             exchange A[i] with A[j]
```

1. If the algorithm is correct prove its correctness by loop invariant. Otherwise argue why it is not correct eg., provide an instance where it fails.
2. State the runtime of the algorithm in asymptotic notation. Justify your answer.

Question 6.2 (0.5 marks)

Consider the following input for RANDOMIZED-QUICKSORT:

12	10	4	2	9	6	5	25	8

What is the probability that:

1. The elements $A[2] = 10$ and $A[3] = 4$ are compared?
2. The elements $A[1] = 12$ and $A[8] = 25$ are compared?
3. The elements $A[4] = 2$ and $A[8] = 25$ are compared?
4. The elements $A[2] = 10$ and $A[7] = 5$ are compared?

Question 6.3 (1 mark)

Prove that the expected runtime of RANDOMIZED-QUICKSORT is $\Omega(n \log n)$.

(HINT: It may be useful to consider how long it takes to compare $n/2$ elements to achieve a lower bound on the runtime.)

Question 6.4 (1 mark)

Draw the decision tree that reflects how SELECTIONSORT sorts $n = 3$ elements. Assume that all elements are mutually distinct.

For convenience here's the pseudocode again:

SELECTION-SORT(A)

```
1: n = A.length
2: for j = 1 to n − 1 do
3:     smallest = j
4:     for i = j + 1 to n do
5:         if A[i] < A[smallest] then smallest = i
6:     exchange A[j] with A[smallest]
```

Question 6.5 (0.5 marks)

What is the smallest possible depth of a leaf in a decision tree for a comparison sort?

Question 6.6 (0.25 marks)

Implement RANDOMIZED-QUICKSORT and solve the 'Yet Another Quicksort' Problem.