Exercise Sheet 5

Handout: Oct 12th — Deadline: Oct 19th, 4pm

Question 5.1 (Marks: 0.25)

Illustrate the operation of QuickSort on the array

4	3	8	2	7	5	1	6

Write down the arguments for each recursive call to QuickSort (e. g. "QuickSort$(A, 2, 5)$") and the contents of the relevant subarray in each step of Partition (see Figure 7.1). Use vertical bars as in Figure 7.1 to indicate regions of values "$\leq x$" and "$> x$". You may leave out elements outside the relevant subarray and calls to QuickSort on subarrays of size 0 or 1.

Question 5.2 (Marks: 0.5)

Prove that deterministic QuickSort(A, p, r) is correct (you can use that Partition is correct since that was proved at lecture).

Question 5.3 (Marks: 0.25)

What is the runtime of QuickSort when the array A contains distinct elements sorted in decreasing order? (Justify your answer)

Question 5.4 (Marks: 0.5)

What value of q does Partition return when all n elements have the same value?
What is the asymptotic runtime (Θ-notation) of QuickSort for such an input? (Justify your answer).

Question 5.5 (Marks: 0.5)

Modify Partition so it divides the subarray in three parts from left to right:

- $A[p \dots i]$ contains elements smaller than x
- $A[i+1 \dots k]$ contains elements equal to x and
- $A[k+1 \dots j-1]$ contains elements larger than x.

Use pseudocode or your favourite programming language to write down your modified procedure Partition' and explain the idea(s) behind it. It should still run in $\Theta(n)$ time for every n-element subarray. Give a brief argument as to why that is the case. Partition' should return two variables q, t such that $A[q \dots t]$ contains all elements with the same value as the pivot (including the pivot itself).

Also write down a modified algorithm QuickSort' that uses Partition' and q, t in such a way that it recurses only on strictly smaller and strictly larger elements.

What is the asymptotic runtime of QuickSort' on the input from Question 5.4?

Question 5.6 (Marks: 0.5)

Implement QuickSort and QuickSort$'$ from Question 5.5.