12 Randomised Algorithms (tms41)

Consider the following algorithm for finding the k-th smallest element of an array $A[1...n]$ consisting of n different natural numbers.

```
QUICK-SELECT(Array A[1...n], position k in [1..n])
   1 If n==1 return A[1]
   2 Pick an index 1 <= j <= n uniformly at random
   3 Let B[1...i] contain the numbers in A less than A[j]
   4 Let B[i+1...n-1] contain the numbers in A greater than A[j]
   5 If i+1==k then return A[j]
   6 else if i>=k then return QUICK-SELECT(B[1...i],k)
   7 else return QUICK-SELECT(B[i+1...n-1],k-i-1)
```

(a) For an input array $A[1 \ldots n]$ of length n, what is the *maximum* (and *minimum*, respectively) number of recursive calls? [4 marks]

Now define the random variable $X_t \in \{0, 1, \ldots, n\}$ to be the length of array A after t recursive calls; so $X_0 := n$. Further, if the algorithm does not make more than t recursive calls, we set $X_t := 0$.

(b) Assuming $n \geq 3$ is odd and $k = \lceil n/2 \rceil$, compute $\mathbb{E}[X_1]$. [4 marks]

(c) Let $n \geq 2$ and $1 \leq k \leq n$ be arbitrary. Assume that there exists $\epsilon > 0$ such that for any $m \geq 1$ and any $t \geq 1$, the inequality $\mathbb{E}[X_t \mid X_{t-1} = m] \leq (1 - \epsilon) \cdot m$ holds.

(i) Which bound can you deduce for any $\mathbb{E}[X_t]$, where $t \geq 1$? [4 marks]

(ii) Prove an upper bound on the maximum number of recursive calls that holds with probability at least $1 - n^{-1}$. [4 marks]

In the following, we will assume that the computation of the two arrays B in lines 3-4 can be performed in $O(n)$ time.

(d) Using earlier parts, how could you express the time complexity of the algorithm? State an upper bound on the expected time complexity. [4 marks]