Algorithms I (CS21003)

Autumn 2010

Assignment 2

A2. (August 3, 2010)

(a) Dynamically allocate a list L of n integers and randomly assign the elements in L such that each elements lies in the closed interval $[a, b]$. Print the input list L.
[User input: n, a, b]

(b) Find $2k + 1$ median elements from L by (recursively) partitioning L. Report the number of comparisons versus n and k.
[User input: $k(\geq 0)$]
Definition: $L_\mu \subset L$ is the sublist of $2k + 1$ median elements if and only if there exist two sublists, L_1 and L_2, such that

i. $L_1 \cup L_\mu \cup L_2 = L$
ii. $|L_1| = \left\lfloor \frac{n-(2k+1)}{2} \right\rfloor$, $|L_2| = \left\lceil \frac{n-(2k+1)}{2} \right\rceil$
iii. $x_1 \leq x_\mu \leq x_2 \; \forall x_1 \in L_1, x_\mu \in L_\mu, x_2 \in L_2$.

Output: File "`roll number_a20.txt`" (Ex: `09CS1001_a20.txt`)
File format: In the order mentioned below, each in a new line.
DON'T print anything extra.

value of n
randomly generated list (tab-delemited) in a single line
value of k
number of (element-to-element) comparisons made throughout the algorithm
L_1 (tab-delemited) in a single line
L_μ (tab-delemited) in a single line
L_2 (tab-delemited) in a single line

Example:
$n = 10, a = 1, b = 6$
$L = \{4, 2, 3, 6, 1, 3, 2, 4, 5, 4\}$
$k = 1 \Rightarrow |L_1| = 3, |L_\mu| = 3, |L_\mu = 4|$.

L_1	L_μ	L_2
2 1 2	3 3 4	4 6 4 5

Output file "`roll number_a20.txt`":

10
4 2 3 6 1 3 2 4 5 4
1
?
2 1 2
3 3 4
4 6 4 5

(c) Use *partitioning by grouping* to solve the above problem.
Report the output in file "`roll number_a21.txt`", having format as in (b).