LaSalle College
Felix Soumpholphakdy

January 28, 2019
Advanced Data Structure: 420-J13-AS

Practice Set 3

A. Implement LOOT-BOX() which returns a string based on the following random percent chances:
    a. 60% chance to return “COMMON”
    b. 30% chance to return “RARE”
    c. 9% chance to return “EPIC”
    d. 1% chance to return “LEGENDARY”

B. Implement PARTITION(A, p, r) which partitions the array A into two subarrays L and R, where every element in L have a value less than or equal to the value of the pivot A[r] and every element in R have a value greater than the value of the pivot A[r].

C. Implement QUICKSORT(A, p, r) which sorts the array A from p to r using the Quicksort algorithm. Use the previously defined PARTITION procedure.

D. Implement QUICKSORT-DESCENDING(A, p, r) which sorts the array A using quicksort but in descending order of elements.

E. Implement RANDOMIZED-PARTITION(A, p, r) which partitions the array A like the previous PARTITION procedure, but instead of picking the last element r as the pivot, it picks a random element.

F. Implement RANDOMIZED-QUICKSORT(A, p, r) which sorts the array A using the new RANDOMIZED-PARTITION procedure.

G. Implement COUNTING-SORT(A, B, k) where A is the array to sort, B is the sorted array (it cannot sort in-place) and k is the array to hold all possible values of elements in A.

H. ***HARD*** Dual-Pivot Quicksort is the implementation of Quicksort in Java 7. Instead of using only one pivot, it uses two pivots (the first and the last element). Write the pseudocode for DUAL-PARTITION(A, p, r) which partitions the array A into three subarrays:
    a. $S_1 = \{x \mid x < A[p]\}$
    b. $S_2 = \{x \mid A[p] < x < A[r]\}$
    c. $S_3 = \{x \mid A[r] < x\}$