Algorithm Analysis and Design

Ghazale Kharadpoor 9513012

November, 2018

Let The List Ziba be an array like M with m members. Then we use SELECT for this array to find the k-th character from the main array of the question. Like RANDOMIZED SELECT at page 216 of CLRS Book, the SELECT algorithm finds the desired element by recursively partitioning the input array. In fact, we know that the worst case time complexity for this foundation of the element is $O(n)$. I use the SELECT Algorithm from the CLRS Book as below:

1. Divide the m elements of the input array into $\left\lfloor \dfrac{m}{5} \right\rfloor$ groups of 5 elements each and at most one group made up of the remaining m mod 5 elements.

2. Find the median of each of the $\left\lceil \dfrac{m}{5} \right\rceil$ groups by first insertion-sorting the elements of each group (of which there are at most 5) and then picking the median from the sorted list of group elements.

3. Use SELECT recursively to find the median x of the $\left\lceil \dfrac{m}{5} \right\rceil$ medians found in step 2. (If there are an even number of medians, then by our convention, x is the lower median.)

We use above algorithm to find the median of medians for array M. In fact, M is array of pairs which contain the the character and its frequency. (As the question is mentioned.) I use SELECT Algorithm from CLRS Book - page 220. In this code, $M.character$ is the characters of given array and $M.value$ is the frequency of the character.

FIND-CHARACTER(M, k)

```
 1  median = SELECT(M, k, 0, m − 1)
 2  arrayA[0...m − 1]
 3  arrayB[0...m − 1]
 4  a = 0
 5  b = 0
 6  if M.lenght == 1
 7       return M[0].character
 8  else
 9       sumOfFreq = 0
10       for i = 0 to m − 1
11            if M[i].character > median.character
12                 sumOfFreq = M[i].value + sumOfFreq
13                 arrayA[a] = M[i]
14                 a = a + 1
15            else
16                 arrayB[b] = M[i]
17                 b = b + 1
18       if sumOfFreq < k and k < sumOfFreq + median.value
19            return median.character
20       else if k <= sumOfFreq
21                 FIND-CHARACTER(arrayA, k)
22       else if k > sumOfFreq + median.value
23                 FIND-CHARACTER(arrayB, k − medOfFreq − median.value)
```

We know that the for loop in the algorithm takes $O(m)$ time. The SELECT Algorithm takes $O(m)$ time in worst case too. Although the recursion part of the above algorithm takes $T(\frac{m}{2})$.

So the time complexity is: $T(m) = T(\frac{m}{2}) + O(m)$

Let's use Master Theorem for proving:

$a = 1, b = 2 \quad \Rightarrow \quad \log_b a = 0$

So this is case 3 which $1.f(\frac{m}{2}) < c.m \quad \Rightarrow \quad c = \frac{3}{4}$

So: $T(m) = \theta(m)$