HOMEWORK 3

COMP3121 - ALGORITHM DESIGN

QUESTION 1

PUNEETH KAMBHAMPATI

z5164647

SOLUTION

If our DNA string is a random string of characters like so

SFSDGJMNSDAKFWEDAFGKNEKSNKJSFGALHNKSLKWQKALDF

The following steps will result in calculating the maximum possible venomous levels that can be acquired from removing characters from the DNA string.

STEP 1.
Find the number of total occurrences each character in the string. This function will be O(n) as we have to traverse the entire string to find the totals.

EXAMPLE:
{
S:4
N:4
A:6
K:4
E:6
}

STEP 2.
We want to deduce the maximum L value we can get from the above string by considering the minimum number of occurrences of the letters in 'SNAKE'. This L value will be the best case scenario for our venemosity level.

EXAMPLE:
L = min{S, N, A, K, E} = 4

STEP 3.
Get the optimal string for the current best case L value. This function would be O(L) as the optimal string is of the length 5*L.

EXAMPLE:
'SSSSNNNNAAAAKKKKEEEE'

STEP 4.
Convert the DNA string we were given into the optimal string acquired in **STEP 3.** One way to do this is to apply the Greedy Algorithm. We know what the optimal character for a given position in the string is.

position 0 to (L-1) : S
position L to (2L -1) : N
Position 2L to (3L -1): A
position 3L to (4L -1): K
Position 4L to (5L -1): E

So, we delete every character until we reach the optimal character for a given index of the string.

STEP 5.
After our attempt at converting the given DNA into the optimal solution, if the result from **STEP 4** is the optimal solution (i.e. length = 5*L) then we found the current best optimal solution and we can move on.

'SSSSNNNNAAAAKKKKEEEE'

If our string from **STEP 4** did not give us the valid optimal string,

'SSSSNNNNA'

we know that the current L value is not an achievable level of venom. However, this does not mean that the venom level is 0 (unless L = 1). We might be able to find a valid DNA string at a lower level of Venom.

STEP 6.
We need to find the next highest level of venom possible. We could try a linear approach to check for all values { L-1, L-2, ... 1, 0} until we find a valid DNA. This would end up being an O(n^2) function.

This can be optimised through Binary search to find the optimal L value and give us O(nlogn).

If a given value FAILS, we simply go to half of current venom level. IF a given value succeeds we go to the average between the failed value and the current value and keep repeating it so.
With the above solution, we can find the optimal level of venom from a DNA string in O(nlogn)