

Dynamic

Video - 78

Programming



Note :- This playlist is only for explanation of Dns & solutions.

See my "DP Concepts & Dns" playlist for understanding DP from scratch...



Facebook
Instagram } → codestorywithMIK

Twitter → cswithMIK



→ codestorywithMIK

2D Memoization

1531. String Compression II

Hint

Hard 1.8K 133 ☆ ↻

Companies

Run-length encoding is a string compression method that works by replacing consecutive identical characters (repeated 2 or more times) with the concatenation of the character and the number marking the count of the characters (length of the run). For example, to compress the string "aabccc" we replace "aa" by "a2" and replace "ccc" by "c3". Thus the compressed string becomes "a2bc3".

Notice that in this problem, we are not adding '1' after single characters.

Given a string s and an integer k . You need to delete **at most** k characters from s such that the run-length encoded version of s has minimum length.

Find the *minimum length of the run-length encoded version of s after deleting at most k characters.*

Example:- $S = "aaabccc"$, $K=2$

Output = 4

$aaa ccc$
 $a3c3$
 1111

"aⁱccdc", $K=2$

($i=0$, $K=2$)

delete-i

Keep-i

"cⁱcdc" ($i=1$, $K=2$)

delete-i

Keep-i

"ccdc", $K=1$

$\Rightarrow C_{(keep)} \Rightarrow 1 + \text{solve}(j+1, K); \text{freq} = 1$

$\Rightarrow "CC"_{(keep)} \Rightarrow 1 + 1 + \text{solve}(j+1, K); \text{freq} = 2$

$\Rightarrow "CC"_{(deleted)} \Rightarrow 1 + 1 + \text{solve}(j+1, K-1)$
(keep CC)

$\Rightarrow "CCC"_{(keep)} \Rightarrow 1 + 1 + \text{solve}(j+1, K-1) \text{freq} = 3$

$\text{Solve}(i, K, s) \{$

$\text{if } (K < 0) \text{ return INT_MAX};$

$\text{if } (i \geq n \text{ || } (n-i) \leq K) \text{ return } 0;$

$\text{int delete-}i = \text{solve}(i+1, K-1, s);$

$\text{int keep-}i = \text{INT_MAX};$

$\text{char ch} = s[i]; \text{ addition} = 0, \text{ deleted} = 0;$

$\text{for } (j = i; j < n \text{ \& deleted} \leq K; j++) \{$

$\text{if } (s[j] == \text{ch}) \{$

$\text{freq}++;$

$\text{if } (\text{freq} == 2 \text{ || } \text{freq} == 10 \text{ || } \text{freq} == 100) \{$

addition++;

} else {

deleted++;

}

Keep-i = min(Keep-i, 1 + addition +
Solve(j+1, K-deleted,
s))

}

return min(delet-i, Keep-i);

}

K=4

ⁱ
"a b c d"

n=4

i=1

n-i=3
