

C3-G 题题解

王家豪

October 18, 2022

G 题题意

给定一个字符串 $T = 1145141919810$ ，输入一个字符串 S ($1 \leq |S| \leq 10^5$)，将 S 切成若干段，分别为 s_1, s_2, \dots, s_m ，将每一段与 T 做比较，统计相同的位置数量记为 $f(s_i, T)$ ，求 $\sum_{i=1}^m f(s_i, T)$ 的最大值

任一字符串的长度不超过 T ，即 $|s_i| \leq |T|$ ，否则可以进一步切分，总是能得到不劣于切分前的结果

1145141919800110
1145141919810

count = 11

1145141919800
1145141919810
110
1145141919810

count = 11 + 2 = 13

核心思路

记 $dp[i]$ 为利用字符串 S 前 i 位进行计算所能得到的最大值, 则有

$$dp[i] = \max_{\max(1, i-13) \leq j < i} dp[j] + cnt(S[j+1:i])$$

11452419198114511	S
1	$cnt = 1$
11	$cnt = 2$
114	$cnt = 1$
1145	$cnt = 0$
11451	$cnt = 2$
114514	$cnt = 5$

核心代码

```
for (int i = 1; i <= n; i++) {  
    for (int l = 1; l <= L; l++) {  
        if (i - l + 1 <= 0) break;  
        int cnt = 0;  
        for (int j = i - l + 1; j <= i; j++) {  
            cnt += s[j] == T[j - (i - l)];  
        }  
        dp[i] = max(dp[i], dp[i - l] + cnt);  
    }  
}
```

时间复杂度: $O(tnL^2)$, OJ 实测 133ms

一些优化

```
for (int i = 0; i <= n; i++) {  
    int cnt = 0;  
    for (int l = 1; l <= L; l++) {  
        if (i + l > n) break;  
        cnt += s[i + l] == T[l];  
        dp[i + l] = max(dp[i + l], dp[i] + cnt);  
    }  
}
```

时间复杂度: $O(tnL)$, OJ 实测 41ms

完整代码

```
#include <stdio.h>
#include <string.h>
#define max(A, B) ((A) > (B) ? (A) : (B))

#define N 100005

char s[N];
char T[] = "x1145141919810";
int dp[N];

int main() {
    int L = strlen(T + 1);
    int tt;
    scanf("%d", &tt);
    while (tt--) {
        scanf("%s", s + 1);
        int n = strlen(s + 1);
        for (int i = 1; i <= n; i++) {
            dp[i] = 0;
        }
        for (int i = 0; i <= n; i++) {
            int cnt = 0;
            for (int l = 1; l <= L; l++) {
                if (i + l > n) break;
                cnt += s[i + l] == T[l];
                dp[i + l] = max(dp[i + l], dp[i] + cnt);
            }
        }
        printf("%d\n", dp[n]);
    }
    return 0;
}
```