

GXUACM2022GreenHand03 题解

A QAQ(枚举)

给定一个字符串 $s(1 \leq |s| \leq 100)$, 问其中不连续的为QAQ的子序列最多有多少个 时间复杂度 $O(|s|)$

思路

只需要维护枚举时前后有多少个没有用过的Q的个数即可

```
front = 0
back = s.count("Q")
res = 0
for i in s:
    if i == 'A':
        res += front * back
    else:
        front += i == 'Q'
        back -= i == 'Q'
print(res)
```

B The Cake Is a Lie(动态规划 思维 贪心?)

给定一个 $n \times m$ 的矩阵以及一个整数 $k(1 \leq n, m \leq 100, 1 \leq k \leq 10^4)$, 问从 $(1, 1)$ 处能否在恰好花费为 k 的情况下进行以下两种操作走到 (n, m) 处, 时间复杂度为 $O(1)$

- 向右移动, 从 $(x, y) \rightarrow (x, y + 1)$, 花费 x
- 向下移动, 从 $(x, y) \rightarrow (x + 1, y)$, 花费 y

思路

- 当进行向右操作时, 下一次向下的操作的花费+1
- 当进行向下的操作时, 下一次向下的操作的花费+1

观察操作的花费可以得出无论何种走法, 最终花费都是相同的, 都等于 $m - 1 + m * (n - 1)$

因此判断 $m - 1 + m * (n - 1)$ 是否等于 k 即可

C Same Differences(思维, 枚举)

给定一个长度为 n 的数组 $a(1 \leq n \leq 2 * 10^5, 1 \leq a_i \leq n)$, 问有多少组 (i, j) 使得 $a_j - a_i == j - i (i < j)$, 时间复杂度为 $O(n)$

思路

易知差值其实只与元素本身及其下标决定, 因此只需要处理元素对下标的差值即可, 差值相同的必定满足式子 $a_j - a_i == j - i (i < j)$, 之后枚举贡献即可

```

Map = dict()
res = 0
for i in range(n):
    a[i] -= i
    if Map.get(a[i]) != None:
        res += Map.get(a[i])
        Map[a[i]] += 1
    else:
        Map[a[i]] = 1
print(res)

```

D I Hate 1111

给定一个数 $n(1 \leq n \leq 10^9)$, 判断 n 能否由11, 111, 1111, ...这些数组成

思路

先观察这些数的特殊性

- $1111 = 11 * 100 + 11$
- $11111 = 111 * 100 + 11$
- $111111 = 111 * 1000 + 11$
-

观察得出除了11和111之外其他的111...都能由11和111来表示

因此我们只需判断 n 能否由11和111组成即可, 可以表示为 $n = 11a + 111b(0 \leq a, 0 \leq b)$

又因为 $111 * 11 == 11 * 111$, 代表111大于11的时候可以将11个111换成111个11, 即可以 b 整除11的部分直接用 $(b / 11) * 11$ 来表示, 所以式子变成了 $n = 11a + 111b(0 \leq a, 0 \leq b \leq 10)$

因此只需要枚举 b 的个数即可

```

for i in range(0, 11):
    if (n - 111 * i) % 11 == 0:
        print("YES")
        return
print("NO")

```

E Plus from Picture(模拟)

给定一个由*和.组成的 $n \times m$ 的矩阵(), 判断矩阵中是否当且仅当只存在一个十字架, 十字架的定义是

- 十字架中间要有*
- 十字架中心点的上下左右四个方向(可以延伸)至少要由一个*

思路

由于十字架是一整个整体, 因此我们只需要找到一个为\$

, 且其上下左右都为的点; 若是找不到则不存在十字架, 若是找得到, 则将整个十字架删除变成.

, 随后只需要在遍历一次整个矩阵, 查找是否还有剩余的*\$即可, 找不到就是YES, 找得到就是NO

G s-palindrome(模拟)

给定一个字符串判断其是否为关于中心对称, 如

- *AHIOMTUVWXY* 就是中心对称图形
- *b*和*d*, *p*和*q*等互为中心对称图形

用map存下每个字符对应的中心对称图形, 之后前后同时遍历判断即可

```
l = 0, r = n - 1
while l <= r:
    if (Map[l] == r):
        continue
    else:
        print("NO")
        return
print("YES")
```

G Vasya and String(双指针)

给定一个由*a*, *b*构成的长度为*n*字符串*s*, 以及一个整数*k* ($1 \leq n \leq 10^5, 0 \leq k \leq n$), 问可以不超过*k*个字符可以得到的最长的连续且完全相同的字串的长度是多少

思路

用双指针遍历分别对*a*, *b*进行贪心并存下maxAns('a')和max('b')并输出较大哪个答案即可即可

```
def value(ch: str) -> int:
    tk = k
    res = l = r = 0
    while l < n and r < n:
        if s[r] == ch:
            res = max(res, r - l + 1)
            r += 1
        else:
            if tk > 0:
                res = max(res, r - l + 1)
                tk -= 1
                r += 1
            else:
                while l < n and s[l] == ch:
                    l += 1
                l += 1
                r += 1
```

```
    return res  
print(max(value("a"), value("b")))
```