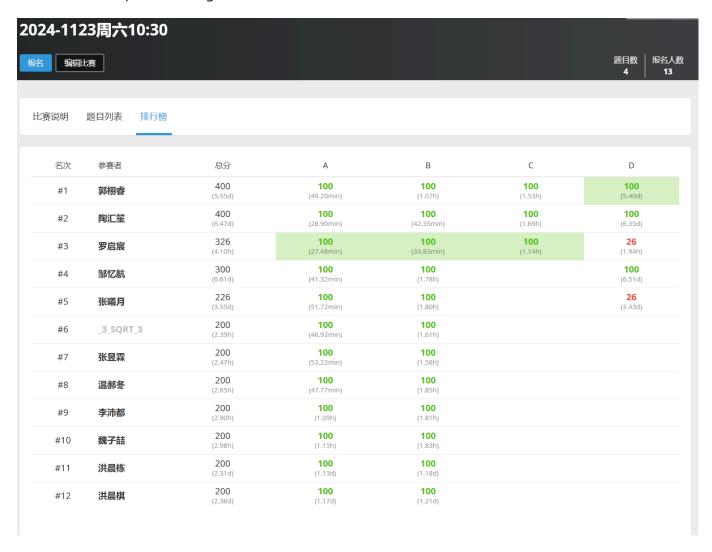
# 重复字符串

## 人员

魏子喆、崔宸赫、陶汇笙、邹忆航、郭栩睿、温郝冬、洪晨栋、洪晨棋、宋吉相、徐本正、李沛都、张昱霖、 王静嘉 到课,罗启宸 线上

## 作业检查

上周作业链接: https://www.luogu.com.cn/contest/215456



## 作业

https://www.luogu.com.cn/contest/217306

# 课堂表现

同学们上课听讲都很认真,今天的第二题,同学们课下要好好复习一下。

# 课堂内容

AT\_abc220\_c Long Sequence

```
#include <bits/stdc++.h>
using namespace std;
typedef long long LL;
const int maxn = 1e5 + 5;
int w[maxn];
int main()
 int n; cin >> n;
  for (int i = 1; i <= n; ++i) cin >> w[i];
  LL x; cin >> x;
  LL sum = 0;
  for (int i = 1; i \le n; ++i) sum += w[i];
  LL k = x / sum;
  LL t = k * sum;
 for (int i = 0; i <= n; ++i) {
   t += w[i];
   if (t > x) {
      cout << k*n+i << endl;</pre>
      break;
  }
  return 0;
}
```

### U504498 字符串解压

```
}
}
return 0;
}
```

#### U510530 珍珠链

```
      string s;

      cin >> s;

      int n = s.size();

      0 ~ n-1

      i: 0~n-1

      j: i~n-1

      检查 i~j 这一段是否符合要求

      统计 i~j 中 R/G 各自的数量

      如果符合要求,求 j-i+1 的最大值
```

```
int maxx = 0;
for (int i = 0; i <= n-1; i++) {
    for (int j = i; j <= n-1; j++) {
        int cnt1=0, cnt2=0;
        for (int k = i; k <= j; k++) {
            if (s[k] == 'R') {
                cnt1++;
            } else {
                  cnt2++;
            }
        }
        if (cnt1 == cnt2) {
            maxx = max(maxx, j-i+1);
        }
    }
}
cout << maxx << endl;</pre>
```

#### P8739 [蓝桥杯 2020 国 C] 重复字符串

```
    字符串长度必须是 k 的倍数
    求 len = n/k, 每 len 个字符为一组
    1, len+1, 2*len+1, ... <=n</li>
    2, len+2, 2*len+2, ... <=n</li>
    3, len+3, 2*len+3, ... <=n</li>
```

```
#include <bits/stdc++.h>
using namespace std;
int cnt[200];
char s[100005];
int main() {
   int k;
    cin >> k;
    cin >> (s+1);
    int n = strlen(s+1);
    if (n%k != 0) {
        cout << -1 << endl;</pre>
        return 0;
    }
    int len = n/k; // 每一组有 len 个
    int sum = 0;
    for (int i = 1; i <= len; i++) { // 以 i 开头
        // 谁和 i 一组:
        // i+len, i+len+len, i+len+len
        for (int j = 'a'; j <= 'z'; j++) {
            cnt[j] = 0;
        for (int j = i; j \leftarrow n; j+=len) {
            cnt[s[j]]++;
        }
        int maxx = 0;
        for (int j = 'a'; j <= 'z'; j++) { // 找 cnt['a'] ~ cnt['z'] 的最大值
            maxx = max(maxx, cnt[j]);
        }
       sum += k - maxx;
    cout << sum << endl;</pre>
```

```
return 0;
}
```