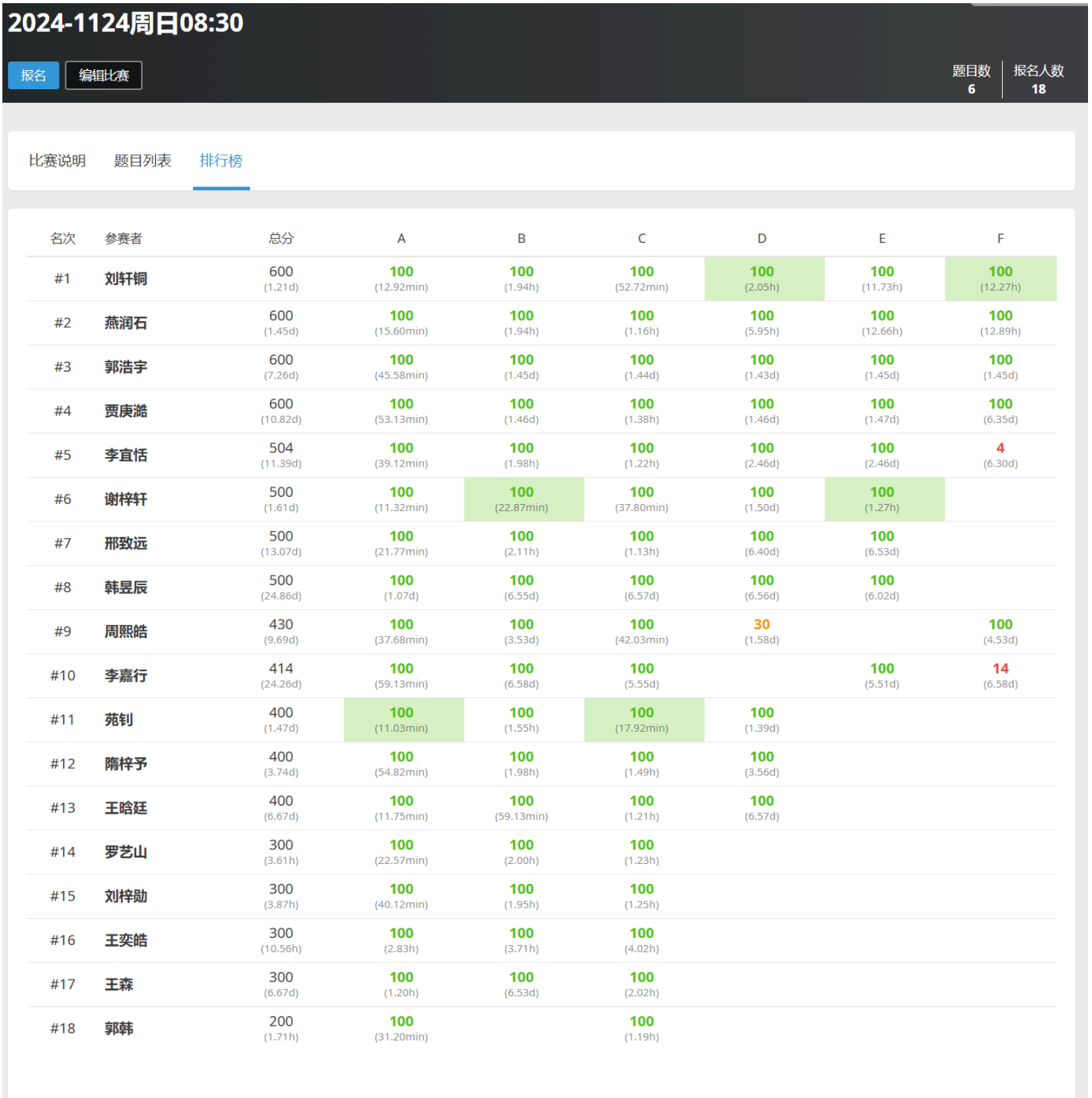
欢乐的跳

人员

刘轩铜、隋梓予、李宜恬、郭浩宇、燕润石、谢梓轩、王森、王晗廷、李嘉行、刘梓勋、罗艺山、邢志远、胡曝辰、王奕皓 到课

上周作业检查

上周作业链接: https://www.luogu.com.cn/contest/215732



2024-1124周日08:30

报名 编辑比赛

题目数 6 | 报名人数 18

比赛说明 题目列表 排行榜

名次	参赛者	总分	A	B	C	D	E	F
#1	刘轩铜	600 (1.21d)	100 (12.92min)	100 (1.94h)	100 (52.72min)	100 (2.05h)	100 (11.73h)	100 (12.27h)
#2	燕润石	600 (1.45d)	100 (15.60min)	100 (1.94h)	100 (1.16h)	100 (5.95h)	100 (12.66h)	100 (12.89h)
#3	郭浩宇	600 (7.26d)	100 (45.58min)	100 (1.45d)	100 (1.44d)	100 (1.43d)	100 (1.45d)	100 (1.45d)
#4	贾庚澔	600 (10.82d)	100 (53.13min)	100 (1.46d)	100 (1.38h)	100 (1.46d)	100 (1.47d)	100 (6.35d)
#5	李宜恬	504 (11.39d)	100 (39.12min)	100 (1.98h)	100 (1.22h)	100 (2.46d)	100 (2.46d)	4 (6.30d)
#6	谢梓轩	500 (1.61d)	100 (11.32min)	100 (22.87min)	100 (37.80min)	100 (1.50d)	100 (1.27h)	
#7	邢致远	500 (13.07d)	100 (21.77min)	100 (2.11h)	100 (1.13h)	100 (6.40d)	100 (6.53d)	
#8	韩昱辰	500 (24.86d)	100 (1.07d)	100 (6.55d)	100 (6.57d)	100 (6.56d)	100 (6.02d)	
#9	周熙皓	430 (9.69d)	100 (37.68min)	100 (3.53d)	100 (42.03min)	30 (1.58d)		100 (4.53d)
#10	李嘉行	414 (24.26d)	100 (59.13min)	100 (6.58d)	100 (5.55d)		100 (5.51d)	14 (6.58d)
#11	苑钊	400 (1.47d)	100 (11.03min)	100 (1.55h)	100 (17.92min)	100 (1.39d)		
#12	隋梓予	400 (3.74d)	100 (54.82min)	100 (1.98h)	100 (1.49h)	100 (3.56d)		
#13	王晗廷	400 (6.67d)	100 (11.75min)	100 (59.13min)	100 (1.21h)	100 (6.57d)		
#14	罗艺山	300 (3.61h)	100 (22.57min)	100 (2.00h)	100 (1.23h)			
#15	刘梓勋	300 (3.87h)	100 (40.12min)	100 (1.95h)	100 (1.25h)			
#16	王奕皓	300 (10.56h)	100 (2.83h)	100 (3.71h)	100 (4.02h)			
#17	王森	300 (6.67d)	100 (1.20h)	100 (6.53d)	100 (2.02h)			
#18	郭韩	200 (1.71h)	100 (31.20min)		100 (1.19h)			

作业

https://www.luogu.com.cn/contest/217538

课堂表现

同学们这节课听讲都很认真，欢乐的跳 这个题很多同学课上没有写完，课下要努力把它写出来。

课堂内容

U510480 奖学金

```cpp
#include <bits/stdc++.h>

using namespace std;

const int maxn = 100 + 5;
int w[maxn];

int main()
{
  int n; cin >> n;
  for (int i = 1; i <= n; ++i) cin >> w[i];
  sort(w+1, w+n+1);
  reverse(w+1, w+n+1);

  for (int i = 1; i <= 2; ++i) cout << w[i] << " " << 500 << endl;
  for (int i = 3; i <= 6; ++i) cout << w[i] << " " << 300 << endl;
  for (int i = 7; i <= 10; ++i) cout << w[i] << " " << 100 << endl;
  for (int i = 11; i <= n; ++i) cout << w[i] << " " << 0 << endl;
  return 0;
}
```

U510481 珠子计算

```cpp
#include <bits/stdc++.h>

using namespace std;

const int maxn = 80 + 5;
int w[maxn];

int main()
{
  int n; cin >> n;
  for (int i = 1; i <= n; ++i) cin >> w[i];
  w[n+1] = w[1];

  int cnt = 0;
  for (int i = 1; i <= n; ++i) {
    if (w[i] != w[i+1]) cnt++;
  }
  cout << cnt << endl;
```

```
    return 0;
}
```

abs, swap, min, max 等方法

```
1. abs: 求绝对值

2. swap: 交换  ->  swap(a, b)

3. min: 取最小值  ->  min(a, b)

4. max: 取最大值  ->  max(a, b)

int maxx = -100000000;
for (int i = 1; i <= n; i++) {
    maxx = max(maxx, a[i]);
}

int minn = 100000000;
for (int i = 1; i <= n; i++) {
    minn = min(minn, a[i]);
}
```

U506961 小鱼的航程

```
#include <iostream>
using namespace std;
int main() {
    int x, n; cin >> x >> n;
    int sum = 0;
    for (int i = 1; i <= n; i++) {
        if (x>=1 && x<=5) {
            sum += 150;
            sum += 100;
        }
        x++;
        if (x == 8) {
            x = 1;
        }
    }
    cout << sum << endl;
    return 0;
}
```

U506949 放大的X

```cpp
#include <iostream>

using namespace std;

int main()
{
    int n;
    cin >> n;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            if (i==j || i+j==n+1) {
                cout << "X";
            }
            else {
                cout << " ";
            }
        }
        cout << endl;
    }
    return 0;
}
```

U504512 选得票多的同学

```cpp
#include <bits/stdc++.h>

using namespace std;

const int maxn = 1000 + 5;
int w[maxn], cnt[maxn];

int main()
{
  int n; cin >> n >> w[1];
  for (int i = 2; i <= n; ++i) w[i] = (w[i-1]*37 + 33031)%n + 1;

  for (int i = 1; i <= n; ++i) cnt[w[i]]++;

  int maxx = 0, id = 0;
  for (int i = 1; i <= n; ++i) {
    if (cnt[i] > maxx) {
      maxx = cnt[i], id = i;
    }
  }
  cout << id << endl;
  return 0;
}
```

P1152 欢乐的跳

```
// 方法一
a[1]         a[2]      a[3]    ....    a[n-1]      a[n]

b[1] = abs(a[1]-a[2])
b[2] = abs(a[2]-a[3])
   ...
b[n-1] = abs(a[n-1]-a[n])

构造 b 数组:
for (int i = 1; i <= n-1; i++) {
    b[i] = abs(a[i] - a[i+1]);
}

如果 b[1]~b[n-1] 这里面正好 1~n-1 全都只出现一次的话

那么对 b 数组排序之后，一定满足:
    b[1] = 1
    b[2] = 2
    b[3] = 3
    ...
    b[n-1] = n-1
```

```
// 方法二
a[1]         a[2]      a[3]    ....    a[n-1]      a[n]

abs(a[1]-a[2])
abs(a[2]-a[3])
...
abs(a[n-1]-a[n])

要确保上面这些值在 1~n-1 之间的每个数出现一次

int cnt[1005];
cnt[abs(a[1]-a[2])]++;
cnt[abs(a[2]-a[3])]++;
...
cnt[abs(a[n-1]-a[n])]++;

如果要满足题意，必须要满足:
    cnt[1] = 1
    cnt[2] = 1
    cnt[3] = 1
    ...
    cnt[n-1] = 1
```

如果 a[1] 是 100000000, a[2] 是 0
但凡出现 abs(a[i] - a[i+1]) 一旦超过 n-1, 一定是错的