

## #A、 小船往返问题

根据公式  $T = \frac{S}{v}$  计算

得

$$T_1 = \frac{S}{v_1 - v_2} + \frac{S}{v_1 + v_2}, T_2 = \frac{2S}{v_1}$$

相减输出即可

注意保留小数位数,可使用如下代码

```
1 | cout << fixed << setprecision(/* 保留的小数位数 */) << (t1 - t2) << "\n";
```

```
1 | printf("%.*lf\n",r,(t1 - t2));
```

## 优化技巧

有些情况下使用 cin/cout 会很慢到超时,有两个优化技巧

1.关闭流图布,在 main 函数最前面添加

```
1 | ios::sync_with_stdio(0);
2 | cin.tie(NULL);
3 | cout.tie(NULL);
```

2.不要使用 endl ,全部使用 "\n"

## 温馨提示

作为过来人,我想说 printf/scanf 和 cin/cout 不要混用

特别是关闭流图布时,会死的很惨

## #B、 简单数独验证

显然这是一道模拟题,直接模拟即可

可以优化代码量的地方有两个

## 1.如何快速判断九个格子是否为1-9

如果现在有一个存有9个数字的 vector ,我们显然可以将它排序后,判断第  $i$  位(从0开始) 的数字是否为  $i + 1$

全部是则合法,否则不合法(你当然可以开标记变量标记,但是我更喜欢用 & 操作)

(或者丢到 set 去重后看容器大小)

```
1  for(int i = 0; i < 9; i ++){
2      sort(row[i].begin(), row[i].end());
3      sort(col[i].begin(), col[i].end());
4      sort(box[i].begin(), box[i].end());
5  }
6  bool temp = 1;
7  for(int i = 0; i < 9; i ++){
8      for(int j = 0; j < 9; j ++){
9          temp &= (row[i][j] == j);
10         temp &= (col[i][j] == j);
11         temp &= (box[i][j] == j);
12     }
13 }
```

## 2.如何快速归纳

对于宫格,不难发现,如果下标从0开始,则点  $(x, y)$  在宫格  $(\lfloor \frac{x}{3} \rfloor, \lfloor \frac{y}{3} \rfloor)$  中,我们对每一行每一列每一宫格单开 vector 即可

最后,送大家两句话

1.十年OI一场空,不开 long long 见祖宗

2.多测不清空,亲人两行泪

## #C、 又是矩阵扩张

直接暴力使用昨天的代码,然后访问点对  $(x, y)$  ,8分送你

(XMas特有的部分分少,不过很多比赛也是这样,要适应)

昨天说过,原来位置为  $(x, y)$  (下标从0开始)的点对,扩张后其左上角的格子为  $(2x, 2y)$

同理,扩张后位于  $(x, y)$  的点对,扩张前位于  $(\lfloor \frac{x}{2} \rfloor, \lfloor \frac{y}{2} \rfloor)$

不难发现反推一轮后问题规模减小,考虑递归处理

令  $dfs(x, y, n)$  表示  $2^n$  的矩阵中,  $(x, y)$  的值

不难发现若  $(x, y)$  不是  $(\lfloor \frac{x}{2} \rfloor, \lfloor \frac{y}{2} \rfloor)$  扩展出的  $2 \times 2$  小矩阵中最左上角的元素,则

$$dfs(x, y, n) = dfs(\lfloor \frac{x}{2} \rfloor, \lfloor \frac{y}{2} \rfloor, n - 1)$$

否则

$$dfs(x, y, n) = !dfs(\lfloor \frac{x}{2} \rfloor, \lfloor \frac{y}{2} \rfloor, n - 1)$$

若  $n = 0$  时,返回  $s$  即可

单次询问  $O(\log(2^n)) = O(n)$ , 总时间复杂度  $O(Tn)$ , 可以通过

## #D、 编排座位

**本题保证只有一解,即不可能出现多种换位方案符合要求**

我们显然可以枚举两个人,交换后检查,如果符合题目要求,输出即可

立刻获得 50pts

否则怎么办?

若记整个矩阵所有数总和为  $sum$ , 不难发现合法时,各行的和与各列的和都为  $\frac{sum}{n}$

进行一次交换后,至多有 1 行的和大于  $\frac{sum}{n}$ ,至多有 1 行的和小于  $\frac{sum}{n}$ ,至多有 1 列的和大于  $\frac{sum}{n}$ ,至多有 1 列的和小于  $\frac{sum}{n}$

我们记第  $i$  行和为  $sumr_i$ ,第  $i$  列和为  $sumc_i$

若将  $sumr_i$  全部丢到 set  $sr$  中去重,则这个  $set.size()$  仅可能为 1 或 3

$sumc_i$  同理,丢到  $sc$  中

**若  $sr.size() == 3 \ \&\& \ sc.size() == 3$**

不难发现交换的两个位置不在同一行不在同一列

记  $sumr_i < \frac{sum}{n}$  的  $i$  为  $minr$ ,同理,得到  $maxr, minc, maxc$

不难发现交换的两个位置为  $(maxr, maxc), (minr, minc)$

**若  $sr.size() == 1$**

得到  $maxc, minc$

枚举每一行,对于第  $i$  行,检查交换  $(i, maxc), (i, minc)$  后是否合法即可

合法就输出

**若  $sc.size() == 1$**

同上,检查每一列交换  $(maxr, j), (minr, j)$

**考虑如何输出**

对于  $(x, y)$  (下标从0开始),其座位号为  $(x \times n) + y + 1$ ,两个座位号比较分别输出小的和大的即可