



华南师范大学
SOUTH CHINA NORMAL UNIVERSITY

软件学院
SCHOOL OF SOFTWARE

2021 年 AK 杯程序设计竞赛

题目分析和解题思路参考

软件协会香农先修班

2021 年 10 月 23 日

第 I 部分

概况

- A: 果冻与天空之后
- B: 白茶与线程博弈
- C: 果冻与阶梯区域
- D: 白茶与三元方程
- E: 果冻与字串得分
- F: 白茶与排名并列
- G: 果冻与卡牌游戏
- H: 白茶与千层之塔
- I: 果冻与千层之塔
- J: 白茶与数位之积

第 II 部分

Problem A. 果冻与天空之后

- 略

第 III 部分

Problem B. 白茶与线程博弈

- 有 m 个踪迹。
- 每一秒先删掉 $7a$ 个，再在剩下里找 $30b$ 个。
- 问能否找到不少于 $\frac{m}{2}$ 个。

关键点

- $m - 7at - 30b(t - 1) \geq 0 \Rightarrow t = \lfloor \frac{m + 30b}{7a + 30b} \rfloor \leq \frac{m + 30b}{7a + 30b}$
- $f = 30b(t - 1) + m - 7at - 30b(t - 1) = m - 7at$
- $t = 0$ 时, 即 $m < 7a$, 第一次干扰就删光了, 直接判 what a pity

完整代码 (C 语言)

```
1 #include <stdio.h>
2 int m, a, b, t, f;
3 int main()
4 {
5     scanf("%d%d%d", &m, &a, &b);
6     t = (m + 30 * b) / (9 * a + 30 * b);
7     f = m - 7 * a * t;
8     if (t && f >= m / 2)
9         printf("I catch you, baicha");
10    else
11        printf("what a pity");
12    return 0;
13 }
```

第 IV 部分

Problem C. 果冻与阶梯区域

- 给定一个坐标 (x, y) ，判断这个坐标在题图中所在的区域。
- 保证坐标不在区域边界。

关键点

- 第一象限上的点所在区域编号只跟 $\min(x, y)$ 的值有关。
- 将点 x 和 y 的值取相反数不影响点的区域编号。
- 使用 `fabs()` 取绝对值，直接判断 $(|x|, |y|)$ 所在的区域编号。
- $(|x|, |y|)$ 所在的区域编号为 $\lceil \min(|x|, |y|) \rceil$ 。
- 上取整可以调用库函数，也可以手写。

注意事项

- 通常使用 `double` 存储浮点数。
- 浮点数需要使用 `fabs()` 取绝对值。
- 输出的应该是一个整数。

完整代码 (C 语言)

```
1 #include <stdio.h>
2 #include <math.h>
3 int main() {
4     double x, y;
5     scanf("%lf%lf", &x, &y);
6     x = fabs(x);
7     y = fabs(y);
8     printf("%.0lf\n", x < y ? ceil(x) : ceil(y));
9 }
```

第 V 部分

Problem D. 白茶与三元方程

- 解关于 x, y, z 的不定方程 $x^3 + y^3 + z^3 = k$ 。
- x, y, z 都是整数且 $1 \leq |x|, |y|, |z| \leq 50$ 。

- 枚举 x, y, z 所有可能的值。
- 判断此时 $x^3 + y^3 + z^3$ 是否等于 k 。

注意事项

- 记得跳过 $x = 0$ 或 $y = 0$ 或 $z = 0$ 的情况。break 或退出程序。
- 在没有符合条件的 x, y, z 时要输出 no solution。
- 可以优化为 $x \leq y \leq z$ 。

核心代码 (C 语言)

```
1 for (int x = -50; x <= 50; x++) {
2     for (int y = x; y <= 50; y++) {
3         for (int z = y; z <= 50; z++) {
4             if (x && y && z && x * x * x + y * y * y + z * z * z == k)
5                 {
6                     printf("%d %d %d\n", x, y, z);
7                     return 0;
8                 }
9         }
10    }
11    printf("no solution");
```

第 VI 部分

Problem E. 果冻与字符串得分

- 每个字母的得分为它在字母表中的位置。
- 给定长为 n 的字符串 a ，字符串第 i 个字母编号为 i ，所有编号为奇数的字母得分的和是 s_1 ，所有编号为偶数的字母的得分的和为 s_2 ，计算 $(s_1 - s_2)/n$ 的值。

- 不使用数组，直接循环碰到 EOF / 非小写字母跳出。
- 分类编号奇偶性并相应作加减，得到 $s_1 - s_2$ 的值。

注意事项

- 调试时 EOF 的输入方式：键盘 / 重定向读文件。
- 转换为浮点数后再做除法。

核心代码 (C 语言)

```
1 int n = 0, p = 0;
2 while (EOF != scanf("%c", &c))
3 {
4     ++n;
5     if (n % 2 == 1)
6         p += c - 'a' + 1;
7     else
8         p -= c - 'a' + 1;
9 }
10 printf("%lf", 1.0 * p / n);
```

- 使用 `strlen()` 获取字符串长度 n 。
- 遍历字符串，分类讨论下标奇偶性并相应作加减，得到 $s_1 - s_2$ 的值。

注意事项

- 题目中的编号从 1 开始，但字符串下标从 0 开始。
- 不要在判断条件中多次调用 `strlen(s)`。
- 转换为浮点数后再做除法。

字符串读入 (C 语言)

```
1 char a[100010]; // 数组建议开大一些
2 scanf("%s", a); // 此处不需要加取地址值符
```

核心代码 (C 语言)

```
1 int n = strlen(a), ans = 0;
2 for (int i = 0; i < n; i++) {
3     ans += (a[i] - 'a' + 1) * (i % 2 ? -1 : 1);
4 }
5 printf("%lf\n", 1.0 * ans / n);
```

第 VII 部分

Problem F. 白茶与排名并列

- 给定已按亲密度降序排列的得分表，求每人的排名。
- 亲密度越高排名越靠前，且同分者赋予相同的排名。

- 如果某人排在表格第一位或他的分数跟位于他上面一行的人分数不同，他的排名就是他在表格中的行数。
- 否则，他的排名就跟位于他上面一行的人排名相同。

注意事项

- 如果你使用其它方式实现排名计算，别忘了考虑所有人分数均相同的情况。
- 需要输出的是排名和姓名，而不是分数。

核心代码 (C 语言)

```
1 m[0] = -1;
2 for (int i = 1; i <= n; i++) {
3     rank[i] = (m[i] != m[i - 1]) ? i : rank[i - 1];
4 }
5 printf("%d\n", rank[n]);
6 for (int i = 1; i <= n; i++) {
7     if (rank[i] <= r) {
8         printf("%d %s\n", rank[i], x[i]);
9     }
10 }
```

第 VIII 部分

Problem G. 果冻与卡牌游戏

- 三名玩家，每人 7 张包含颜色和数字两种属性的牌。
- 轮到自己时都会出从左往右第一张与参照牌同色或同数字的牌。
- 打出去的牌会变成新的参照牌。
- 连续两人无法出牌则第三人出手中最左边的牌。
- 问谁能获得胜利。

关键点

- 只有轮到了自己而且出牌机会还没使用才可出牌。
 - 只有当某张牌还没出过才可出这张牌。
 - 只有前面两个人没出牌或牌的任一属性与参照牌相同才可出这张牌。
 - 出了牌后必须更新参照牌的属性。
 - 出了牌后还需要判断玩家手中还有没有牌。
 - 要判断玩家本轮是否没有出牌。
-
- 按照题目要求进行模拟。
 - 颜色判断可以优化为只判断首字母。

核心代码 (C 语言)

```
1 int flag = 1, cur = 0;
2 for (int i = 0; i < 7; i++) {
3     if (flag && h[pos][i] && (nocnt == 2 || s[pos][i] == refs || n[pos]
4         ][i] == refn)) {
5         refs = s[pos][i]; refn = n[pos][i]; // 更新参照牌属性
6         h[pos][i] = nocnt = flag = 0;
7     }
8     cur += h[pos][i]; // 统计玩家手中剩余牌的数量
9 }
10 if (cur == 0) {
11     puts(pos ? (pos == 1 ? "lr580" : "dayuanx") : "bobby"); break;
12 }
13 nocnt += flag; pos = (pos + 1) % 3;
```

第 IX 部分

Problem H. 白茶与千层之塔

- 第 k 层有通往第 $k - 1$ 层和第 $k - 2$ 层的电梯。
- 如果 $k \bmod 3 = 1$ ，还有通往第 $k - 3$ 层的电梯。
- 求第 a 层到第 b 层的方案数。
- $b = 1, a \leq 20$ 。

关键点

- 用递归 DFS 模拟乘电梯过程，枚举所有选项。
- 模拟走到 1 层时，结束，并方案数自增。
- 注意不要越界走到小于 1 层的地方。

核心代码 (C 语言)

```
1 void dfs(int x) //当前在第k层
2 {
3     if (x == b) //到达了目标地
4     {
5         s++; //方案数+1
6         return;
7     }
8     if (x - 1 >= 1) //能走才走x-1
9         dfs(x - 1);
10    if (x - 2 >= 1)
11        dfs(x - 2);
12    if (x % 3 == 1 && x - 3 >= 1)
13        dfs(x - 3);
14 }
```

第 X 部分

Problem I. 果冻与千层之塔

- 第 k 层有通往第 $k - 1$ 层和第 $k - 2$ 层的电梯。
- 如果 $k \bmod 3 = 1$ ，还有通往第 $k - 3$ 层的电梯。
- 求第 a 层到第 b 层的方案数。
- $a - b \leq 10^6$ 。

关键点

- 方案数跟始末楼层之差 $a - b$ 的值有关。
- 方案数跟 $a \bmod 3$ 的值有关。
- 设 $dp_{x,y}$ 为在 x 层开始，往下走 y 层的方案数。
- 使用加法原理和乘法原理推导公式：
$$dp_{x,y} = \begin{cases} dp_{x-1,y-1} + dp_{x-2,y-2} + dp_{x-3,y-3}, & x \bmod 3 = 1 \\ dp_{x-1,y-1} + dp_{x-2,y-2}, & x \bmod 3 \neq 1 \end{cases}$$
- 计算初始值 $dp_{x,0} = dp_{x,1} = 1, dp_{x,2} = 2$
- 优化得 x 只取 $[0, 2]$
- 取模公式

核心代码 (C 语言)

```
1 for (int i = 0; i < 3; ++i) //初始值
2     dp[i][0] = dp[i][1] = 1, dp[i][2] = 2;
3 for (int j = 3; j <= 1000000; ++j) //要先第二维再第一维, 注意顺序
4 {
5     for (int i = 0; i < 3; ++i)
6     {
7         if (i % 3 == 1)
8             dp[i][j] = ((dp[(i + 2) % 3][j - 1] + dp[(i + 1) % 3][j -
9                 2]) % mod + dp[i][j - 3]) % mod;
10        else
11            dp[i][j] = (dp[(i + 2) % 3][j - 1] + dp[(i + 1) % 3][j -
12                2]) % mod;
```

第 XI 部分

Problem J. 白茶与数位之积

- $f(k)$ 为十进制下 k 的所有数位的乘积。
- 解关于 l, r 的不定方程 $x = \prod_{i=l}^r f(i)$ 。
- l, r 都是整数且 $0 \leq l \leq r \leq 10^y - 1$ 。

关键点

- 一旦区间中任意数位出现了 0，数位积就是 0。
 - 连续的十个数中必有一个数的个位是 0。
 - 要使位数最少，多个 2, 3 可以进行合并，必定不取 1。
-
- 特判 $x = 0$ 的情况。
 - 令 $l = 10k + l', r = 10k + r', 1 \leq l' \leq r' \leq 9$ 。
 - $f(k) = \left(\frac{x}{\sum_{i=l'}^{r'} i}\right)^{\frac{1}{r'-l'+1}}$ ，如果 $(\sum_{i=l'}^{r'} i) \nmid x$ 在此时的 l', r' 下无解。
 - 将 $\frac{x}{\sum_{i=l'}^{r'} i}$ 分解为 $2^{c_2} \cdot 3^{c_3} \cdot 5^{c_5} \cdot 7^{c_7}$ ，如果 $(r' - l' + 1) \nmid c_j$ 在此时的 l', r' 下无解。
 - 枚举 6 的数量，先贪心取 8, 9，再贪心取 4，再贪心取其它数。

- 如果对题目等仍有任何疑问，欢迎现场提问。

- Thanks!



谢谢

软件协会香农先修班 · 2021 年 AK 杯程序设计竞赛