

2022 CCF 非专业级软件能力认证模拟赛 - 入门组

2022/8/4

(请选手务必仔细阅读本页内容)

题目名称	字符串问题	公约数	数独游戏	入侵攻击
题目类型	传统型	传统型	传统型	传统型
英文题目名称	str	gcd	sudoku	attack
输入文件名	str.in	gcd.in	sudoku.in	attack.in
输出文件名	str.out	gcd.out	sudoku.out	attack.out
输出文件名	1s	1s	1s	1s
内存上限	512M	512M	512M	512M
测试点数目	20	10	5	10
每个测试点分值	5	10	20	10
附加样例文件	有	有	有	有
结果比较方式	全文比较	全文比较	全文比较	全文比较

注意事项:

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C/C++ 中函数 `main()` 的返回类型必须是 `int`，程序正常结束时的返回值必须是 `0`。

1 字符串

(str.cpp/c)

【题目描述】

有 n 个字符串，每个字符串都有一个分值，现在我们要询问以某个字符结尾的字符串中分值第 k 大的字符串是哪个？同时对于分值相同的字符串，先读入的更大。

【输入输出格式】

【输入格式】

从文件 str.in 中读入数据。

第一行两个正整数 n, m

接下来 n 行，每行一个字符串，保证由小写字母组成，长度不超过 50，然后一个整数，表示这个字符串的分值。

接下来 m 行，每行一个字母，表示询问名字以某个字母结尾的字符串，然后一个正整数 k 表示询问排名为 k 大的字符串。

【输出格式】

输出到文件 str.out 中。

m 行，每个询问输出一行，表示对应字符串的名字。如果以该字母结尾的字符串中排名为 k 大的字符串不存在，输出 “Orz YYR tq!”

【样例】

【样例 1 输入】

```
5 2
aaa 1
aa 2
a 3
ab 3
bb 4
b 2
a 4
```

【样例 1 输出】

```
ab
Orz YYR tq!
```

【数据范围与提示】

对于 5% 的数据: $n \leq 10, m \leq 1000$ 。

对于 20% 的数据: $n, m \leq 1000$

对于所有数据 $n, m \leq 10^5, k \leq n$, 保证名字长度小于等于 50

2 公约数

(gcd.cpp/c/pas)

【题目描述】

给你一个 n 我们要计算 n 以内的三元组的最大公约数的和。

```
1 for (int i = 1; i <= n; ++i)
2   for (int j = 1; j <= n; ++j)
3     for (int k = 1; k <= n; ++k)
4       ans += gcd(i, j, k)
```

code sample

$gcd(a, b, c)$ 表示计算三个数的最大公约数。

【输入输出格式】

【输入格式】

输入文件 gcd.in
一行一个正整数 n

【输出格式】

输出文件 gcd.out
只有一行一个整数表示答案

【样例】

【样例 1 输入】

2

【样例 1 输出】

9

【数据范围与提示】

对于 40% 的据有 $n \leq 200$ 。
对于 100% 的据有 $n \leq 1000$ 。

3 数独游戏

(sudoku.cpp/c/pas)

【题目描述】

陈老师最近在玩数独，但是他好像不太聪明的样子，因此他想请聪明的你帮他完成这些数独。

陈老师给了你若干个数独，你需要把每一个 9×9 的数独补充完整，使得数独中每行、每列、每个 3×3 的九宫格内数字 1 - 9 均恰好出现一次。

由于数独问题网上参考代码太多，陈老师要求你独立完成，别去抄代码。

【输入输出格式】

【输入格式】

从文件 sudoku.in 中读入数据

第一行一个整数 T 表示测试数据组数

接下来 T 行，每行包含 81 个字符，代表数独的 81 个格内数据（顺序总体由上到下，同行由左到右）。

每个字符都是一个数字（1-9）或一个.（表示尚未填充）。

你可以假设输入中的每个数独都只有一个解决方案。

【输出格式】

输出到文件 sudoku.out

一共 T 行，每一行数据，代表填充完全后的数独

【样例】

【样例 1 输入】

```
2
4.....8.5.3.....7.....2.....6.....8.4.....1.....6.3.7.5..2.....1.4.....
.....52..8.4.....3...9...5.1...6..2..7.....3.....6...1.....7.4.....3.
```

【样例 1 输出】

```
417369825632158947958724316825437169791586432346912758289643571573291684164875293
416837529982465371735129468571298643293746185864351297647913852359682714128574936
```

【数据范围与提示】

对于 20% 的数据, $T = 1$

对于 60% 的数据, $T \leq 3$

对于 100% 的数据, $T \leq 20$

4 入侵攻击

(attack.cpp/c/pas)

【题目描述】

公元 2089 年，邪恶的滑滑梯星人向银河系发起总攻

为了抵抗滑滑梯星人的入侵，行星防御理事会 (PDC) 付出很大代价绘制出了一份以滑滑梯总部为原点的星际地图，其中包含了 n 个滑滑梯星人可能入侵的星球的三维坐标。

而在之后的几次战役中 PDC 弄清楚了滑滑梯星人的进攻模式：

- 从总部 $(0, 0, 0)$ 驾驶飞船出发
- 每次移动时，如果飞船的曲率引擎的空间翘曲能力为 R ，则与当前飞船所在星球欧氏距离小于等于 R 的星球都可以被直接到达
- 设上步中可到达的还未被占领的星球（不包括飞船现在所在的星球）数量为 k ，则滑滑梯星人会将飞船复制为一模一样的 $k + 1$ 份，其中一艘留在原地占领该星球，其余 k 艘分别前往这 k 个可到达的星球

通过捕获的几艘飞船，PDC 大致推测出了滑滑梯星人曲率引擎的常用功率，现在他们想知道在飞船空间翘曲能力为 $d_1, d_2 \dots d_m$ 的情况下，滑滑梯星人分别能占领多少个星球。

【输入输出格式】

【输入格式】

从文件 attack.in 中读入数据

第一行两个数 n 和 m

以下 n 行每行三个整数 x_i, y_i, z_i ，表示每个可能被入侵的星球的三维坐标

以下 m 行每行一个整数 d_i ，表示 m 次对滑滑梯飞船空间翘曲能力的假设

【输出格式】

输出到文件 attack.out 中

共 m 行，每行一个整数，分别表示在每种假设下滑滑梯星人能占领的星球总数。

注意：滑滑梯总部不算占领的星球

【样例】

【样例 1 输入】

```
5 5
2 0 3
0 4 4
5 1 3
3 7 6
```

4 0 1
2
3
4
5
6

【样例 1 输出】

0
0
3
5
5

【数据范围与提示】

对于 30% 的数据, $n \leq 10$ $m \leq 10$;

对于 60% 的数据, $n \leq 200$, $m \leq 2000$;

对于 100% 的数据, $n \leq 1000$, $m \leq 500000$, $0 \leq x_i, y_i, z_i, d_i \leq 10^9$, 且 x_i 、 y_i 、 z_i 、 d_i 均为整数。

数据存在梯度。在所有数据中均匀分布着 50% 的数据, 满足 $m = 1$ 。

提示

三维欧氏距离计算公式: 假设两点坐标分别为 (x_1, y_1, z_1) 和 (x_2, y_2, z_2) , 则他们之间的欧氏距离为 $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$ 。

样例解释

当 $R \leq 3$ 时, 飞船不能从总部到达任何星球, 所以能占领的星球数为 0。

当 $R = 4$ 时, 飞船可以从总部到达 1 号星球 (距离为 $\sqrt{13}$), 从 1 号星球可以到达 3 号和 5 号星球 (距离分别为 $\sqrt{10}$ 和 $2\sqrt{2}$), 但从总部和这些星球这 4 个点都无法到达其他 2 个星球, 所以能占领的星球数为 3。

当 $R = 5$ 时, 飞船可以从总部到达 1 号和 5 号星球 (距离分别为 $\sqrt{13}$ 和 $\sqrt{17}$), 从 1 号星球到达 2 号星球和 3 号星球 (距离分别为 $\sqrt{21}$ 和 $\sqrt{10}$), 从 2 号星球到达 4 号星球 (距离为 $\sqrt{22}$), 所以能占领的星球数为 5, 即所有星球都可以被占领。

显然当 $R = 6$ 时同样可占领所有星球。