

CSP-J 模拟赛

????.??.??

| 题目名称 | 美丽数 | 比赛 | 他会输出啥 | 游览计划 |
|----------|--------|--------|--------|--------|
| 输入/输出文件名 | beauty | gaming | output | tour |
| 时间限制 | 1s | 1s | 1s | 1s |
| 空间限制 | 512 MB | 512 MB | 512 MB | 512 MB |
| 测试点数目 | 10 | 20 | 20 | 20 |
| 题目类型 | 传统题 | 传统题 | 传统题 | 传统题 |

注意事项

1. 无需建立子文件夹。
2. 文件名（包括程序名和输入输出文件名）必须使用英文小写。
3. 结果比较方式为忽略行末空格、文末回车后的全文比较。
4. 程序可使用的栈空间大小与该题内存空间限制一致。
5. 编译选项： `-O2 -std=c++14 -lm` 。
6. 时限在 std 的 2 倍以上，所以理论上不存在卡常的情况。

美丽数(beauty)

题目描述

小B 对数字有着独到的见解，他认为一个数是美丽数当且仅当这个数的相邻两位都不相同。

例如 12345、12121 是美丽数，而 123321、11 不是美丽数。

现在小B想知道，对于所有满足以下条件的数中，最小的那个数是什么？如果不存在这样的美丽数，则输出 -1。

- 对于 $i(0 \leq i \leq 9)$ ，这个数中是 i 的位数一共有 a_i 个。
- 这个数不能有前导 0，即 012 是不合法的。

有多组测试点，每组测试点现在给你 $a_0 \sim a_9$ ，请你回答小B的问题。

输入格式

第一行包含一个整数 T 。

接下来 T 行，每行 10 个整数， $a_0 \sim a_9$ ，含义如题。

输出格式

输出 T ，每行一个整数，表示答案。

样例

输入

```
1 5
2 5 0 0 0 0 0 0 0 0 5
3 6 0 0 0 0 0 0 0 0 5
4 0 1 0 2 0 0 0 0 0 0
5 3 0 0 0 2 0 0 1 0 0
6 0 1 2 3 4 5 6 7 8 9
```

输出

```
1 1
2 -1
3 313
4 404070
5 123234345454565656767676787897979898989898989
```

数据范围

对于 100% 的数据，保证： $T \leq 10, 1 \leq n, a_i \leq 10^5$ ，其中 $n = \sum_{i=0}^9 a_i$ 。

| 测试点编号 | 数据范围 | 特殊性质 |
|--------|---------------|------|
| 1 ~ 2 | $n \leq 10$ | 无 |
| 3 ~ 4 | $n \leq 10^3$ | 无 |
| 5 ~ 6 | 无限制 | A |
| 7 ~ 8 | 无限制 | B |
| 9 ~ 10 | 无限制 | 无 |

- A : 保证最多只有 2 个 $a_i > 0$ 。
- B : 保证最多只有 3 个 $a_i > 0$ 。

比赛(gaming)

题面描述

一年一度的校园歌手大赛开始了，今年一共有 n 个选手参与了比赛，分别编号为 $1 \sim n$ ，每个人的唱功可以用数字 a_i 来量化，保证 $a_1 \sim a_n$ 互不相同。

今年赛事委员会决定让比赛变得更刺激一点，于是采用了以下的赛制：

- 编号为 $2d + 1$ 的选手和编号为 $2d + 2$ 的选手会通过合唱来PK (d 为整数)，唱功更高的选手（即 a_i 更大的选手）获胜，晋级下一轮。如果 n 为奇数，则编号为 n 的选手轮空，直接晋级。这一轮所有淘汰的选手，其最终名次就是第 $\lceil \frac{n}{2} \rceil + 1$ 名。
- 所有晋级的选手重新编号 $1 \sim \lceil \frac{n}{2} \rceil$ ，继续下一轮。
- 直到剩下一个选手，Ta就是歌手大赛的冠军，其名次也就是第 1 名。

唱功不错的小 B 却惨遭一轮游，原因是第一轮遇到的对手最后获得了冠军。

所以他为了向赛事委员会说明这个比赛规则的不合理性，他想对于每一个 i ，求出如果初始编号为 i 的选手能在第一次被淘汰的那场比赛获胜晋级，他最终能得到什么名次。

输入格式

第 1 行包含一个正整数 n ，表示选手数量。

第 2 行包含 n 个正整数 a_1, a_2, \dots, a_n 表示每一位选手的唱功。

输出格式

包含一行 n 个整数，含义如题。

样例一

输入

```
1 10
2 3 9 6 12 15 24 30 18 27 21
```

输出

```
1 4 3 4 2 4 2 1 4 1 2
```

样例解释

第一轮选手 [3, 9, 6, 12, 15, 24, 30, 18, 27, 21]

第二轮 [9, 12, 24, 30, 27]

第三轮 [12, 30, 27]

第四轮 [30, 27]

第五轮 [30]

对于选手 2，他第一次被淘汰在第二轮，若获胜，则第三轮选手是 [9, 30, 27]，所以选手 2 最后的名次变成了 3。

数据范围

对于所有数据 $n \leq 5 \times 10^5$, $a_i \leq 10^9$ ，并且互不相同。

| 测试点 | $n \leq$ | 特殊性质 |
|---------|----------|------|
| 1 ~ 4 | 10 | 无 |
| 5 ~ 10 | 10^3 | 无 |
| 11 ~ 14 | 无限制 | A |
| 15 ~ 20 | 无限制 | 无 |

A : 保证 a_i 递增。

他会输出啥(output)

题目描述

小 B 最近迷上了Python这门语言，但连C++都搞不太清楚的他自然也搞不太懂Python。

这天，小B遇到了一个Python程序，但发现运行了很久也没有输出，现在他想请教你有什么办法能够快速得到结果。

这个Python程序的形式如下：

```
1 ans=0
2 for A in range(a,b,c):
3     for B in range(d,e,f):
4         ans=ans+B
5 print(ans)
```

其中 A,B 一定是小写字母 a~z 中的一个，并且保证 A 不等于 B 。

保证 a,b,c,f 一定是一个绝对值 $\leq 10^6$ 的整数。

保证 d,e 要么是一个绝对值 $\leq 10^6$ 的整数要么就是 A 所代表的字母。

下面是关于 `for A in range(a,b,c)` 的解释：



在Python中，`for i in range(a, b, c):` 是一个 `for` 循环的结构，它使用 `range(a, b, c)` 函数来生成一个序列，然后迭代这个序列中的每个元素。下面是详细的解释：

- `for`：这是Python中的一个关键字，用于开始一个 `for` 循环。
- `i`：这是一个变量，用于在每次迭代中存储 `range()` 函数生成的序列中的当前元素。在循环的每次迭代中，`i` 会自动更新为序列中的下一个值。
- `range(a, b, c)`：这是 `range` 函数的调用，用于生成一个整数序列。这个序列从 `a` 开始，到 `b-1` 结束（不包括 `b`），并且每个数之间的间隔（步长）是 `c`。

需要注意的是，`c` 可以是正数也可以是负数，但绝对不能为0，否则会引发 `ValueError`。如果 `c` 是正数，那么 `a` 必须小于 `b` 才能生成序列；如果 `c` 是负数，那么 `a` 必须大于 `b`。

- `:`：冒号表示 `for` 循环的开始，并且在冒号后面的是循环体，即每次迭代要执行的代码块。

- 循环体：这是在 `for` 循环中缩进的代码块，它包含了一个或多个语句，这些语句将在每次迭代时执行。

例如，考虑以下 `for` 循环：

```
1 for i in range(2, 10, 2):
2     for j in range(10, i, -4):
3         print(i,j)
```

这个循环将执行以下操作：

1. `range(2, 10, 2)` 生成一个序列：2, 4, 6, 8。
2. `for` 循环开始迭代这个序列。
3. 在第一次迭代中，`i` 被赋值为2，然后执行循环体中的 `for j in range(10, 2, -4):`，打印出：

```
1 2 10
2 2 6
```

4. 在第二次迭代中，`i` 被赋值为4，然后执行循环体中的 `for j in range(10, 4, -4):`，打印出：

```
1 4 10
2 4 6
```

5. 在第三次迭代中，`i` 被赋值为6，然后执行循环体中的 `for j in range(10, 6, -4):`，打印出：

```
1 6 10
```

6. 在第四次迭代中，`i` 被赋值为8，然后执行循环体中的 `for j in range(10, 8, -4):`，打印出：

```
1 8 10
```

输入格式

一共五行，形如：

```
1 ans=0
2 for A in range(a,b,c):
3     for B in range(d,e,f):
4         ans=ans+B
5 print(ans)
```

输出格式

一行一个整数，表示答案，保证答案存在。

样例一

输入

```
1 ans=0
2 for i in range(10,1,-2):
3     for j in range(i,10,3):
4         ans=ans+j
5 print(ans)
```

输出

```
1 49
```

数据范围

对于所有数据， A, B 一定是小写字母 $a \sim z$ 中的一个，并且保证 A 不等于 B 。

保证 a, b, c, f 一定是一个绝对值 $\leq 10^6$ 的整数。

保证 d, e 要么是一个绝对值 $\leq 10^6$ 的整数要么就是 A 所代表的字母。

| 测试点 | a, b 范围 | d, e 范围 | d, e 是否一定为整数 | c, f 范围 |
|-------|--------------------------|--------------------------|----------------|------------|
| 1 ~ 2 | $0 \leq b$ 且 $b = a + 1$ | $0 \leq e$ 且 $e = d + 1$ | 是 | $c, f > 0$ |
| 3 ~ 4 | $0 \leq a, b$ | $0 \leq e$ 且 $e = d + 1$ | 是 | $c, f > 0$ |

| 测试点 | a,b 范围 | d,e 范围 | d,e 是否一定为整数 | c,f 范围 |
|---------|-------------------------|-----------------------------|-------------|------------|
| 5 ~ 6 | 无限制 | $0 \leq e$ 且 $e = d + 1$ | 是 | 无限制 |
| 7 ~ 8 | $0 \leq a, b \leq 10^3$ | $0 \leq d, e \leq 10^3$ | 是 | $c, f > 0$ |
| 9 ~ 10 | $0 \leq a, b \leq 10^3$ | $0 \leq d, e \leq 10^3$ | 否 | $c, f > 0$ |
| 11 ~ 13 | $ a , b \leq 10^3$ | $ d , e \leq 10^3$ | 否 | 无限制 |
| 14 ~ 15 | $0 \leq a, b$ | $0 \leq d, e$ | 是 | $c, f > 0$ |
| 16 ~ 17 | 无限制 | 无限制 | 否 | $c, f > 0$ |
| 18 ~ 20 | 无限制 | 无限制 | 否 | 无限制 |

游览计划(tour)

题目描述

小 B 写大模拟题写烦了，于是来到了一个旅游景点散散心。

这个旅游景点的地图共有 n 处地点，在这些地点之间连有 m 条双向道路。旅客从一个景点前往另一个景点需要乘坐景点提供的旅游大巴，旅游大巴会按照经过道路条数最少的路线行驶。

小 B 购买的景区套票让小 B 只能游览这 n 处地点中的 4 处，而且小 B 喜欢浏览沿途的风景，所以小 B 希望选出这 4 处不同的景点 a, b, c, d ，使 $a \rightarrow b, b \rightarrow c, c \rightarrow d$ 这三条旅游大巴行驶路线经过的道路数量总和最多。

你只需要输出最多的道路数量是多少。

输入格式

第一行两个整数 n, m 。

接下来 m 行，每行两个整数 x_i, y_i ，表示第 i 条道路连接了第 x_i 和 y_i 处地点。

输出格式

共一行一个整数，表示答案。

样例一

输入

```
1 6 10
2 1 2
3 1 3
4 3 4
5 2 5
6 4 6
7 2 6
8 1 4
9 5 3
10 5 4
11 3 2
```

输出

16

样例解释

一种方案是选择 3, 6, 5, 1 参观，经过 $2 + 2 + 2 = 6$ 条道路。

数据范围

对于所有数据 $n \leq 4000, m \leq 5000, 1 \leq x_i, y_i \leq n$ ，保证无重边，自环。

| 测试点 | 数据范围 |
|---------|------------------------|
| 1 ~ 3 | $n \leq 10$ |
| 4 ~ 8 | $n \leq 50$ |
| 9 ~ 10 | $m = \frac{n(n-1)}{2}$ |
| 11 ~ 15 | $n \leq 400$ |
| 16 ~ 20 | 无限制 |