

1、组队竞赛

网上举办了一次编程比赛,参加比赛的有 $3*n$ 个选手,每个选手都有一个水平值 a_i 。现在要将这些选手进行组队,一共组成 n 个队伍,即每个队伍3人.牛牛发现队伍的水平值等于该队伍队员中第二高水平值。

例如:

一个队伍三个队员的水平值分别是(3,3,3), 那么队伍的水平值是3

一个队伍三个队员的水平值分别是(3,2,3), 那么队伍的水平值是3

一个队伍三个队员的水平值分别是(1,5,2), 那么队伍的水平值是2

为了让比赛更有看点,牛牛想安排队伍使所有队伍的水平值总和最大。

如样例所示:

如果牛牛把6个队员划分到两个队伍

如果方案为:

team1:{1,2,5}, team2:{5,5,8}, 这时候水平值总和为7.

而如果方案为:

team1:{2,5,8}, team2:{1,5,5}, 这时候水平值总和为10.

没有比总和为10更大的方案,所以输出10

输出一个整数表示所有队伍的水平值总和最大值.

输入描述:

输入的第一行为一个正整数 $n(1 \leq n \leq 10^5)$

第二行包括 $3*n$ 个整数 $a_i(1 \leq a_i \leq 10^9)$,表示每个参赛选手的水平值.

输出描述:

输出一个整数表示所有队伍的水平值总和最大值.

示例:

输入

```
2
5 2 8 5 1 5
```

输出

```
10
```

2、倒置字符串

将一句话的单词进行倒置, 标点不倒置。比如 I like beijing. 经过函数后变为: beijing. like I

输入描述:

每个测试输入包含1个测试用例: I like beijing. 输入用例长度不超过100

输出描述:

依次输出倒置之后的字符串,以空格分割

示例:

输入

```
I like beijing
```

输出

```
beijing like I
```

3、字符串中找出连续最长的数字串

读入一个字符串str，输出字符串str中的连续最长的数字串

输入描述：

个测试输入包含1个测试用例，一个字符串str，长度不超过255。

输出描述：

在一行内输出str中里连续最长的数字串。

示例：

输入

```
abcd12345ed125ss123456789
```

输出

```
123456789
```

4、数组中出现次数超过一半的数字

给一个长度为 n 的数组，数组中有一个数字出现的次数超过数组长度的一半，请找出这个数字。

例如输入一个长度为9的数组[1,2,3,2,2,2,5,4,2]。由于数字2在数组中出现了5次，超过数组长度的一半，因此输出2。

数据范围： $n \leq 50000$ ，数组中元素的值 $0 \leq val \leq 10000$

要求：空间复杂度： $O(1)$ ，时间复杂度 $O(n)$

输入描述：

先输入n，然后输入n个数，保证有解。

输出描述：

输出超过数组长度一半的数字

示例1：

输入：

```
9
1 2 3 2 2 2 5 4 2
```

输出:

2

示例2:

输入:

7
3 3 3 3 2 2 2

输出:

3

示例3:

输入:

1
1

输出:

1

5、计算糖果

A,B,C三个人是好朋友,每个人手里都有一些糖果,我们不知道他们每个人手上具体有多少个糖果,但是我们知道以下的信息:

A - B, B - C, A + B, B + C. 这四个数值.每个字母代表每个人所拥有的糖果数.

现在需要通过这四个数值计算出每个人手里有多少个糖果,即A,B,C。这里保证最多只有一组整数A,B,C满足所有题设条件。

输入描述:

输入为一行,一共4个整数,分别为A - B, B - C, A + B, B + C, 用空格隔开。范围均在-30到30之间(闭区间)。

输出描述:

输出为一行,如果存在满足的整数A, B, C则按顺序输出A, B, C, 用空格隔开,行末无空格。如果不存在这样的整数A, B, C, 则输出No

示例:

输入:

1 -2 3 4

输出

2 1 3

6、进制转换

给定一个十进制数M，以及需要转换的进制数N。将十进制数M转化为N进制数

输入描述：

输入为一行，M(32位整数)、N($2 \leq N \leq 36$)，以空格隔开。

输出描述：

为每个测试实例输出转换后的数，每个输出占一行。如果N大于9，则对应的数字规则参考16进制（比如，10用A表示，等等）

示例：

输入：

```
7 2
```

输出：

```
111
```

7、统计回文

“回文串”是一个正读和反读都一样的字符串，比如“level”或者“noon”等等就是回文串。花花非常喜欢这种拥有对称美的回文串，生日的时候她得到两个礼物分别是字符串A和字符串B。现在她非常好奇有没有办法将字符串B插入字符串A使产生的字符串是一个回文串。你接受花花的请求，帮助她寻找有多少种插入办法可以使新串是一个回文串。如果字符串B插入的位置不同就考虑为不一样的办法。

例如：

A = “aba”，B = “b”。这里有4种把B插入A的办法：

* 在A的第一个字母之前: “baba” 不是回文

* 在第一个字母’a’之后: “abba” 是回文

* 在字母’b’之后: “abba” 是回文

* 在第二个字母’a’之后 “abab” 不是回文

所以满足条件的答案为2

输入描述：

每组输入数据共两行。第一行为字符串A 第二行为字符串B 字符串长度均小于100且只包含小写字母

输出描述：

输出一个数字，表示把字符串B插入字符串A之后构成一个回文串的方法数

示例：

输入：

```
aba
b
```

输出：

8、字符串转成整数

将一个字符串转换成一个整数，要求不能使用字符串转换整数的库函数。数值为 0 或者字符串不是一个合法的数值则返回 0

数据范围：字符串长度满足 $0 \leq n \leq 100$

进阶：空间复杂度 $O(1)$ ，时间复杂度 $O(n)$

注意：

①字符串中可能出现任意符号，出现除 +/- 以外符号时直接输出 0

②字符串中可能出现 +/- 且仅可能出现在字符串首位。

输入描述：

输入一个字符串,包括数字字母符号,可以为空

输出描述：

如果是合法的数值表达则返回该数字，否则返回0

示例1：

输入：

```
+2147483647
```

输出：

```
2147483647
```

示例2：

输入：

```
1a33
```

输出：

```
0
```

9、Fibonacci数列

Fibonacci数列是这样定义的：

$F[0] = 0$

$F[1] = 1$

for each $i \geq 2$: $F[i] = F[i-1] + F[i-2]$

因此，Fibonacci数列就形如：0, 1, 1, 2, 3, 5, 8, 13, ...，在Fibonacci数列中的数我们称为Fibonacci数。

给你一个N，你想让其变为一个Fibonacci数，每一步你可以把当前数字X变为X-1或者X+1，现在给你一个数N求最少需要多少步可以变为Fibonacci数。

输入描述:

输入为一个正整数 $N(1 \leq N \leq 1,000,000)$

输出描述:

输出一个最小的步数变为Fibonacci数

示例:

输入:

15

输出:

2

10、合法括号序列判断

给定一个字符串A，请输出YES或NO判断其中的小括号"()"是否正确匹配。

输入描述:

输入为一个字符串，长度满足 $0 \leq n \leq 1000$

输出描述:

输出YES或NO

示例1:

输入:

((()())

输出:

YES

示例2:

输入:

((()a+((x-c)*d)((b))))

输出:

YES

示例3:

输入:

((()())())

输出:

NO

11、求最小公倍数

正整数A和正整数B 的最小公倍数是指 能被A和B整除的最小的正整数，设计一个算法，求输入A和B的最小公倍数。

数据范围: $1 \leq a, b \leq 100000$

示例:

输入:

5 7

输出:

35

12、另类加法

输入两个int A和B，输出它们的和，但不能使用算术运算符 (+ - * / %)

示例:

输入:

1 2

输出:

3

13、求路径总数

请计算 $n \times m$ 的棋盘格子 (n 为横向的格子数, m 为竖向的格子数) 从棋盘左上角出发沿着边缘线从左上角走到右下角, 总共有多少种走法, 要求不能走回头路, 即: 只能往右和往下走, 不能往左和往上走。

注: 沿棋盘格之间的边缘线行走

数据范围: $1 \leq n, m \leq 8$

示例:

输入:

2 2

输出:

6

14、井字棋

井字棋是一种棋盘大小为3*3的游戏。现在规定左上角位置为(0,0)，右下角位置为(2,2)，现在输入某三个同色棋子的坐标x和y，判断是否可以赢得游戏。

输入描述:

输入共三行，每行两个数字n和m表示横纵坐标 ($0 \leq n, m \leq 2$)

输出描述:

输出YES或NO

示例:

输入:

```
0 0
1 0
2 0
```

输出:

YES

15、密码强度等级

密码按如下规则进行计分，并根据不同的得分为密码进行安全等级划分。

一、密码长度:

5 分: 小于等于4 个字符

10 分: 5 到7 字符

25 分: 大于等于8 个字符

二、字母:

0 分: 没有字母

10 分: 密码里的字母全都是小（大）写字母

20 分: 密码里的字母符合“大小写混合”

三、数字:

0 分: 没有数字

10 分: 1 个数字

20 分: 大于1 个数字

四、符号:

0 分: 没有符号

10 分: 1 个符号

25 分: 大于1 个符号

五、奖励（只能选符合最多的那一种奖励）：

2 分: 字母和数字

3 分: 字母、数字和符号

5 分: 大小写字母、数字和符号

最后的评分标准:

>= 90: 非常安全

>= 80: 安全 (Secure)

>= 70: 非常强

>= 60: 强 (Strong)

>= 50: 一般 (Average)

>= 25: 弱 (Weak)

>= 0: 非常弱 (Very_Weak)

对应输出为:

VERY_SECURE

SECURE

VERY_STRONG

STRONG

AVERAGE

WEAK

VERY_WEAK

请根据输入的密码字符串，进行安全评定。

注:

字母: a-z, A-Z

数字: 0-9

符号包含如下:

!"#\$%&'()*+,-./ (ASCII码: 0x21~0x2F)

::<=>?@ (ASCII码: 0x3A~0x40)

[]^_` (ASCII码: 0x5B~0x60)

{|}~ (ASCII码: 0x7B~0x7E)

输入描述:

一行，一个字符串（字符串长度 ≤ 300 ）

输出描述:

密码等级

示例:

输入:

```
38$@NoNoN
```

输出:

```
VERY_SECURE
```

解释：长度大于等于8个字符，得25分；大小写字母都有所以得20分；有两个数字，所以得20分；包含大于1符号，所以得25分；由于该密码包含大小写字母、数字和符号，所以奖励部分得5分，经统计得该密码的密码强度为25+20+20+25+5=95分。

16、求最大连续bit数

求一个int类型数字对应的二进制数字中1的最大连续数，例如3的二进制为00000011，最大连续2个1

数据范围：数据组数： $1 \leq t \leq 5$ ， $1 \leq n \leq 500000$

进阶：时间复杂度： $O(\log n)$ ，空间复杂度： $O(1)$

输入描述：

输入一个int类型数字

输出描述：

输出转成二进制之后连续1的个数

示例：

输入：

15

输出：

4

17、二进制插入

给定两个32位整数n和m，同时给定i和j，将m的二进制数位插入到n的二进制的第j到第i位,保证n的第j到第i位均为零，且m的二进制位数小于等于i-j+1，其中二进制的位数从0开始由低到高。

输入描述：

四个数字：n, m, i, j。 ($0 \leq n + m \leq 31$)

输出描述：

输出转成二进制之后连续1的个数

示例：

输入：

1024 19 2 6

输出：

1100

18、查找组成一个偶数最接近的两个素数

任意一个偶数（大于2）都可以由2个素数组成，组成偶数的2个素数有很多种情况，本题目要求输出组成指定偶数的两个素数差值最小的素数对。

数据范围：输入的数据满足 $4 \leq n \leq 1000$

输入描述：

输入一个大于2的偶数

输出描述：

从小到大输出两个素数

示例：

输入：

20

输出：

7 13

19、计算日期到天数转换

根据输入的日期，计算是这一年的第几天。

保证年份为4位数且日期合法。

进阶：时间复杂度： $O(n)$ ，空间复杂度： $O(1)$

输入描述：

输入一行，每行空格分割，分别是年，月，日

输出描述：

输出是这一年的第几天

示例：

输入：

2012 12 31

输出：

366

20、查找输入整数二进制中1的个数

输入一个正整数，计算它在二进制下的1的个数。

输入描述：

输入一个整数 ($1 \leq n \leq 2^{31}-1$)

输出描述：

计算整数二进制中1的个数

示例：

输入：

5

输出：

2