

Assignment #4: 国庆节月考

钟明衡2300011505

2023/9/26

说明:

- 1) 国庆节月考: AC8。摸底题目都在“练习”里面, 按照数字题号能找到, 可以重新提交。作业中提交自己最满意版本的代码和截图。
- 2) 请把每个题目解题思路(可选), 源码Python, 或者C++/C(已经在Codeforces/Openjudge上AC), 截图(包含Accepted, 学号), 填写到下面作业模版中(推荐使用 typora <https://typoraio.cn>, 或者用 word)。AC 或者没有AC, 都请标上每个题目大致花费时间。
- 3) 提交时候先提交pdf文件, 再把md或者doc文件上传到右侧“作业评论”。Canvas需要有同学清晰头像、提交文件有pdf、作业评论有md或者doc。
- 4) 如果不能在截止前提交作业, 请写明原因。

编程环境

操作系统: Windows_NT x64 10.0.19045

Python编程环境: Visual Studio Code 1.76.1

C/C++编程环境: Visual Studio Code 1.76.1

1. 必做题目

02701: 与7无关的数

math, <http://cs101.openjudge.cn/practice/02701>

一个正整数,如果它能被7整除,或者它的十进制表示法中某一位上的数字为7,则称其为与7相关的数.现求所有小于等于 n ($n < 100$) 的与7无关的正整数的平方和.

输入

输入为一行,正整数 n ($n < 100$)

输出

输出一行, 包含一个整数, 即小于等于 n 的所有与7无关的正整数的平方和。

样例输入

1 | 21

样例输出

来源

计算概论05

【钟明衡，物理学院，2023年秋】

思路：

由于n上限仅为100，可以直接列出和7有关的数，然后累加时跳过这些数即可

代码

```
1 n = int(input())
2 l = [7, 14, 17, 21, 27, 28, 35, 37, 42, 47, 49, 56, 57, 63,
3     67, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 84, 87, 91, 97, 98]
4 ans = 0
5 for i in range(1, n+1):
6     if i not in l:
7         ans += i**2
8 print(ans)
9
```

代码运行截图

#41484220提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
n = int(input())
l = [7, 14, 17, 21, 27, 28, 35, 37, 42, 47, 49, 56, 57, 63,
    67, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 84, 87, 91, 97, 98]
ans = 0
for i in range(1, n+1):
    if i not in l:
        ans += i**2
print(ans)
```

基本信息

#: 41484220

题目: 02701

提交人: 23n2300011505(12号娱乐选手)

内存: 3636kB

时间: 34ms

语言: Python3

提交时间: 2023-10-05 21:38:47

02712: 细菌繁殖

math, <http://cs101.openjudge.cn/practice/02712>

一种细菌的繁殖速度是每天成倍增长。例如：第一天有10个，第二天就变成20个，第三天变成40个，第四天变成80个，……。现在给出第一天的日期和细菌数目，要你写程序求出到某一天的时候，细菌的数目。

输入

第一行有一个整数n，表示测试数据的数目。其后n行每行有5个整数，整数之间用一个空格隔开。第一个数表示第一天的月份，第二个数表示第一天的日期，第三个数表示第一天细菌的数目，第四个数表示要求的那一天的月份，第五个数表示要求的那一天的日期。已知第一天和要求的一天在同一年并且该年不是闰年，要求的一天一定在第一天之后。数据保证要求的一天的细菌数目在长整数（long）范围内。

输出

对于每一组测试数据，输出一行，该行包含一个整数，为要求的一天的细菌数。

样例输入

```
1 | 2
2 | 1 1 1 1 2
3 | 2 28 10 3 2
```

样例输出

```
1 | 2
2 | 40
```

来源

2005~2006医学部计算概论期末考试

【钟明衡，物理学院，2023年秋】

思路：

直接计算目标日期之间相差的天数，然后将初始细菌数乘以2的这个天数次幂即可

代码

```
1 | n = int(input())
2 | days = [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31]
3 | for i in range(n):
4 |     m1, d1, ans, m2, d2 = map(int, input().split())
5 |     count = d2-d1
6 |     for j in range(m1-1, m2-1):
7 |         count += days[j]
8 |     print(ans*(2**count))
9 |
```

代码运行截图

状态: **Accepted**

源代码

```
n = int(input())
days = [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31]
for i in range(n):
    m1, d1, ans, m2, d2 = map(int, input().split())
    count = d2-d1
    for j in range(m1-1, m2-1):
        count += days[j]
    print(ans*(2*count))
```

基本信息

#: 41484331

题目: 02712

提交人: 23n2300011505(12号娱乐选手)

内存: 3560kB

时间: 29ms

语言: Python3

提交时间: 2023-10-05 21:45:52

02753: 菲波那契数列

math, <http://cs101.openjudge.cn/practice/02753>

菲波那契数列是指这样的数列: 数列的第一个和第二个数都为1, 接下来每个数都等于前面2个数之和。给出一个正整数a, 要求菲波那契数列中第a个数是多少。

输入

第1行是测试数据的组数n, 后面跟着n行输入。每组测试数据占1行, 包括一个正整数a(1 ≤ a ≤ 20)

输出

输出有n行, 每行输出对应一个输入。输出应是一个正整数, 为菲波那契数列中第a个数的大小

样例输入

```
1 | 4
2 | 5
3 | 2
4 | 19
5 | 1
```

样例输出

```
1 | 5
2 | 1
3 | 4181
4 | 1
```

【钟明衡, 物理学院, 2023年秋】

思路:

直接将斐波那契数列的前20项存到列表中, 然后输出即可

代码

```
1 n = int(input())
2 l = [1, 1]
3 for i in range(18):
4     l.append(l[-1]+l[-2])
5 for i in range(n):
6     print(l[int(input())-1])
7
```

代码运行截图

#41484396提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```
n = int(input())
l = [1, 1]
for i in range(18):
    l.append(l[-1]+l[-2])
for i in range(n):
    print(l[int(input())-1])
```

基本信息

#: 41484396

题目: 02753

提交人: 23n2300011505(12号娱乐选手)

内存: 3556kB

时间: 33ms

语言: Python3

提交时间: 2023-10-05 21:49:52

02810: 完美立方

bruteforce, <http://cs101.openjudge.cn/practice/02810>

形如 $a^3 = b^3 + c^3 + d^3$ 的等式被称为完美立方等式。例如 $12^3 = 6^3 + 8^3 + 10^3$ 。编写一个程序，对任给的正整数N ($N \leq 100$)，寻找所有的四元组 (a, b, c, d)，使得 $a^3 = b^3 + c^3 + d^3$ ，其中 a,b,c,d 大于 1，小于等于N，且 $b \leq c \leq d$ 。

输入

一个正整数N ($N \leq 100$)。

输出

每行输出一个完美立方。输出格式为：

Cube = a, Triple = (b,c,d)

其中a,b,c,d所在位置分别用实际求出四元组值代入。

请按照a的值，从小到大依次输出。当两个完美立方等式中a的值相同，则b值小的优先输出、仍相同则c值小的优先输出、再相同则d值小的先输出。

样例输入

1 | 24

样例输出

```
1 Cube = 6, Triple = (3,4,5)
2 Cube = 12, Triple = (6,8,10)
3 Cube = 18, Triple = (2,12,16)
4 Cube = 18, Triple = (9,12,15)
5 Cube = 19, Triple = (3,10,18)
6 Cube = 20, Triple = (7,14,17)
7 Cube = 24, Triple = (12,16,20)
```

【钟明衡，物理学院，2023年秋】

思路：

数据量不大，直接枚举即可

利用 $b \leq c \leq d$ 可以略微优化

代码

```
1 n = int(input())
2 for a in range(2, n+1):
3     for b in range(2, a):
4         for c in range(b, a):
5             for d in range(c, a):
6                 if a**3 == b**3+c**3+d**3:
7                     print('Cube = %d, Triple = (%d,%d,%d)' % (a, b, c, d))
8
```

代码运行截图

#41484601提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
n = int(input())
for a in range(2, n+1):
    for b in range(2, a):
        for c in range(b, a):
            for d in range(c, a):
                if a**3 == b**3+c**3+d**3:
                    print('Cube = %d, Triple = (%d,%d,%d)' % (a, b, c, d))
```

基本信息

#: 41484601
题目: 02810
提交人: 23n2300011505(12号娱乐选手)
内存: 3624kB
时间: 3747ms
语言: Python3
提交时间: 2023-10-05 22:03:39

04138: 质数的和与积

math, <http://cs101.openjudge.cn/practice/04138>

两个质数的和是S，它们的积最大是多少？

输入

一个不大于10000的正整数S，为两个质数的和。

输出

一个整数，为两个质数的最大乘积。数据保证有解。

样例输入

```
1 | 50
```

样例输出

```
1 | 589
```

来源

《奥数典型题举一反三（小学五年级）》(ISBN 978-7-5445-2882-5) 第三章 第二讲 例1

【钟明衡，物理学院，2023年秋】

思路：

先写一个函数来判断是否为质数

然后从 $n/2$ 到2来判断是否是两个质数的和，从数学上可知，第一个判断出的结果就是最大的

代码

```
1 def check(n):
2     if n == 2 or n == 3 or n == 5:
3         return True
4     elif n % 2 == 0:
5         return False
6     else:
7         for i in range(3, int(n**0.5)+1, 2):
8             if n % i == 0:
9                 return False
10            return True
11
12
13 n = int(input())
14 for i in range(int(n/2)+1, 1, -1):
15     if check(i):
16         if check(n-i):
17             print(i*(n-i))
18             break
19
```

状态: Accepted

源代码

```
def check(n):
    if n == 2 or n == 3 or n == 5:
        return True
    elif n % 2 == 0:
        return False
    else:
        for i in range(3, int(n**0.5)+1, 2):
            if n % i == 0:
                return False
        return True

n = int(input())
for i in range(int(n/2)+1, 1, -1):
    if check(i):
        if check(n-i):
            print(i*(n-i))
            break
```

基本信息

#: 41484886
题目: 04138
提交人: 23n2300011505(12号娱乐选手)
内存: 4244kB
时间: 30ms
语言: Python3
提交时间: 2023-10-05 22:31:43

04146: 数字方格

math, <http://cs101.openjudge.cn/practice/04146>

a1	a2	a3
----	----	----

如上图，有3个方格，每个方格里面都有一个整数 a_1 , a_2 , a_3 。已知 $0 \leq a_1, a_2, a_3 \leq n$ ，而且 $a_1 + a_2$ 是2的倍数， $a_2 + a_3$ 是3的倍数， $a_1 + a_2 + a_3$ 是5的倍数。你的任务是找到一组 a_1 , a_2 , a_3 ，使得 $a_1 + a_2 + a_3$ 最大。

输入

一行，包含一个整数 n ($0 \leq n \leq 100$)。

输出

一个整数，即 $a_1 + a_2 + a_3$ 的最大值。

样例输入

1 | 3

样例输出

1 | 5

思路：

直接枚举即可

代码

```
1 n = int(input())
2 ans = 0
3 for a1 in range(n, -1, -1):
4     for a2 in range(n, -1, -1):
5         for a3 in range(n, -1, -1):
6             if a1+a2+a3 > ans:
7                 if (a1+a2) % 2 == 0 and (a2+a3) % 3 == 0 and (a1+a2+a3) % 5 ==
0:
8                     ans = a1+a2+a3
9 print(ans)
10
```

代码运行截图

#41484982提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
n = int(input())
ans = 0
for a1 in range(n, -1, -1):
    for a2 in range(n, -1, -1):
        for a3 in range(n, -1, -1):
            if a1+a2+a3 > ans:
                if (a1+a2) % 2 == 0 and (a2+a3) % 3 == 0 and (a1+a2+a3)
                    ans = a1+a2+a3
print(ans)
```

基本信息

#: 41484982

题目: 04146

提交人: 23n2300011505(12号娱乐选手)

内存: 3620kB

时间: 138ms

语言: Python3

提交时间: 2023-10-05 22:46:05

2. 选做题目

02746: 约瑟夫问题

implementation, <http://cs101.openjudge.cn/practice/02746>

约瑟夫问题：有 n 只猴子，按顺时针方向围成一圈选大王（编号从 1 到 n ），从第 1 号开始报数，一直数到 m ，数到 m 的猴子退出圈外，剩下的猴子再接着从 1 开始报数。就这样，直到圈内只剩下一只猴子时，这个猴子就是猴王，编程求输入 n ， m 后，输出最后猴王的编号。

输入

每行是用空格分开的两个整数，第一个是 n ，第二个是 m ($0 < m, n \leq 300$)。最后一行是：

0 0

输出

对于每行输入数据（最后一行除外），输出数据也是一行，即最后猴王的编号

样例输入

```
1 6 2
2 12 4
3 8 3
4 0 0
```

样例输出

```
1 5
2 1
3 7
```

【钟明衡，物理学院，2023年秋】

思路：

直接模拟就可以了，利用取整来实现闭环的效果

注意使用pop以减少时间复杂度

代码

```
1 while True:
2     n, m = map(int, input().split())
3     if n == 0:
4         break
5     elif m == 1:
6         print(n)
7         continue
8     l = []
9     for i in range(n):
10         l.append(i+1)
11     i = m-1
12     while n > 1:
13         if i > n-1:
14             i = i % n
15         n -= 1
16         l.pop(i)
17         i += m-1
18     print(l[0])
19
```

代码运行截图

状态: Accepted

源代码

```
while True:
    n, m = map(int, input().split())
    if n == 0:
        break
    elif m == 1:
        print(n)
        continue
    l = []
    for i in range(n):
        l.append(i+1)
    i = m-1
    while n > 1:
        if i > n-1:
            i = i % n
        n -= 1
        l.pop(i)
        i += m-1
    print(l[0])
```

基本信息

#: 41485371

题目: 02746

提交人: 23n2300011505(12号娱乐选手)

内存: 3520kB

时间: 41ms

语言: Python3

提交时间: 2023-10-05 23:54:22

CF1364A: A. XXXXX

brute force/data structures/number theory/two pointers, 1200, <https://codeforces.com/problemset/problem/1364/A>

Ehab loves number theory, but for some reason he hates the number x . Given an array a , find the length of its longest subarray such that the sum of its elements **isn't** divisible by x , or determine that such subarray doesn't exist.

An array a is a subarray of an array b if a can be obtained from b by deletion of several (possibly, zero or all) elements from the beginning and several (possibly, zero or all) elements from the end.

Input

The first line contains an integer t ($1 \leq t \leq 5$) — the number of test cases you need to solve. The description of the test cases follows.

The first line of each test case contains 2 integers n and x ($1 \leq n \leq 10^5$, $1 \leq x \leq 10^4$) — the number of elements in the array a and the number that Ehab hates.

The second line contains n space-separated integers a_1, a_2, \dots, a_n ($0 \leq a_i \leq 10^4$) — the elements of the array a .

Output

For each testcase, print the length of the longest subarray whose sum isn't divisible by x . If there's no such subarray, print -1 .

Example

input

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

output

1	2
2	3
3	-1

Note

In the first test case, the subarray [2,3] has sum of elements 5, which isn't divisible by 3.

In the second test case, the sum of elements of the whole array is 6, which isn't divisible by 4.

In the third test case, all subarrays have an even sum, so the answer is -1.

【钟明衡，物理学院，2023年秋】

思路：

重要的是子序列之和与x的整除，因此将所有余数加起来，讨论以下情况：

如果所有数都能整除x，则不存在，输出-1

如果所有数之和不能整除x，则答案就是n

如果能整除，但是不是每个数都能整除，就找出最靠前或者最靠后的一个不能整除x的数，这个数所在离两头最近的次序要从答案的n中减去

代码

```

1  t = int(input())
2  for i in range(t):
3      n, x = map(int, input().split())
4      l = list(map(int, input().split()))
5      count = 0
6      ans = False
7      for a in l:
8          if a % x != 0:
9              ans = True
10             count += a % x
11     if ans:
12         if count % x == 0:
13             for j in range(1, n):
14                 if l[j-1] % x != 0 or l[n-j] % x != 0:
15                     n -= j

```

```

16         break
17     print(n)
18 else:
19     print('-1')
20

```

代码运行截图

General									
#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged	
226767026	Practice: MinghengZhong	1364A - 15	Python 3	Accepted	265 ms	18308 KB	2023-10-05 19:15:30	2023-10-05 19:15:31	★ Compare

→ Source
Copy

```

t = int(input())
for i in range(t):
    n, x = map(int, input().split())
    l = list(map(int, input().split()))
    count = 0
    ans = False
    for a in l:
        if a % x != 0:
            ans = True
            count += a % x
    if ans:
        if count % x == 0:
            for j in range(1, n):
                if l[j-1] % x != 0 or l[n-j] % x != 0:
                    n -= j
                    break
            print(n)
        else:
            print('-1')

```

3. 学习总结和收获

数据量较少的时候，可以完全使用计算机思维，把所有情况都尝试一遍

但是如果数据较多，这样很可能会超时，因此需要优化算法，比如剪枝，或者采用数学方法

如果使用的算法不对从而TLE，就算从Python改成Pypy或者C++，也会超时，只有改变算法才有可能AC