

# 211122循环结构练习题

---

## 211122循环结构练习题

### 一、基础回顾

#### 1.1 知识点for-in,range()

- 1.偶数
- 2.奇数
- 3.连续奇数的和
- 4.乘法表

#### 1.2 while 循环 break、continue

- 1.递增序列

### 二、提高

- 1.约数
- 2.最大数和它的位置

### 三、巩固

- 1.菱形

## 一、基础回顾

---

### 1.1 知识点for-in,range()

---

#### 1.偶数

---

编写一个程序，输出 1 到 100 之间（包括 1 和 100）的全部偶数。

```
for i in range(0,101,2):  
    print(i)
```

#### 2.奇数

---

输入一个整数 X，输出 1 到 X 之间（包括 1 和 X）的全部奇数。

```
for i in range(1,101,2):  
    print(i)
```

#### 3.连续奇数的和

---

给定两个整数 X 和 Y，输出在他们之间（不包括 X 和 Y）的所有奇数的和。

不用函数的写法

```

a=int(input())
b=int(input())
if a>b:
    a,b=b,a
    s = 0
    for i in range(a + 1, b):
        if i % 2 == 1:
            s = s + i
    print(s)
else:
    s = 0
    for i in range(a + 1, b):
        if i % 2 == 1:
            s = s + i
    print(s)

```

## 用函数的写法

#在不确定x和y的大小前提下，先做判断，如果a>b、交换，再操作求奇数和。

```

def jishuhe(a,b):
    s = 0
    for i in range(a + 1, b):
        if i % 2 == 1:
            s = s + i
    return s
if __name__ == '__main__':
    a=int(input())
    b=int(input())
    if a>b:
        a,b=b,a
        rst= jishuhe(a,b)
        print(rst)
    else:
        rst= jishuhe(a, b)
        print(rst)

```

## 4.乘法表

输入一个整数 N，输出 N 的乘法表，如下：

```
1 x N = N
2 x N = 2N
...
10 x N = 10N
```

### 输入格式

一个整数 N。

### 输出格式

输出 N的乘法表，具体形式参照输出样例。

### 数据范围

$1 < N < 1000$

### 输入样例：

```
140
```

### 输出样例：

```
1 x 140 = 140
2 x 140 = 280
3 x 140 = 420
4 x 140 = 560
5 x 140 = 700
6 x 140 = 840
7 x 140 = 980
8 x 140 = 1120
9 x 140 = 1260
10 x 140 = 1400
```

### python 参考代码

```
# 乘法表
n=int(input())
for i in range(1,11):
    print("%d * %d = %d"%(i,n,i*n))
```

### C++参考代码

```
#include <iostream>
using namespace std;

int main()
{
    int n;
    cin >> n;
    for(int i=1; i<=10; i++)
    {
        printf("%d x %d = %d\n", i, n, i*n);
    }
}
```

## 1.2 while 循环 break、continue

### 1.递增序列

读取一系列的整数 X，对于每个 X，输出一个 1,2,...,X的序列。

#### 输入格式

输入文件中包含若干个整数，其中最后一个为 0，其他的均为正整数。

每个整数占一行。

对于输入的正整数，按题目要求作输出处理。

对于最后一行的整数 0，不作任何处理。

#### 输出格式

对于每个输入的正整数 X，输出一个从 1 到 X 的递增序列，每个序列占一行。

#### 数据范围

$1 \leq X \leq 100$

#### 输入样例：

```
5
10
3
0
```

#### 输出样例：

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

## Python参考代码

```
while True:
    n = int(input())
    if not n:
        break;
    else:
        for i in range (n):
            print (i+1, end = " ")
        print()
```

## C++参考代码

```
#include<iostream>
#include<cstdio>
using namespace std;
int main()
{
    int a;
    while(1)
    {
        cin>>a;
        if(a==0)
            break;
        for(int i=1;i<=a;i++)
            cout<<i<<" ";
        cout<<endl;
    }
    return 0;
}
```

# 二、提高

## 1.约数

输入一个整数 N，按照从小到大的顺序输出它的全部约数。

输入格式

一个整数  $N$ 。

### 输出格式

输出全部约数，每个约数占一行。

### 数据范围

$1 \leq N \leq 1000$

### 输入样例：

```
6
```

### 输出样例：

```
1
2
3
6
```

### C++参考代码

```
#include <iostream>
#include <algorithm>

using namespace std;

int main()
{
    int n;

    cin >> n;

    for (int i = 1; i <= n; i++)
    {
        if (n % i == 0)

            cout << i << endl;
    }

    return 0;
}
```

## 2.最大数和它的位置

---

给定 100 个整数，请你找出其中最大的数字，以及它的输入位置（位置从 1 开始）。

### 输入格式

共 100 行，每行包含一个整数。

### 输出格式

第一行输出最大的数字。

第二行输出该数字的输入位置。

### 数据范围

$1 \leq \text{输入数字} \leq 50000$ ,  
保证输入数字互不相同。

### 输入样例：

```
22229
48558
24992
4755
11923
...
20213
```

### 输出样例：

```
48558
2
```

### 参考代码：

#### Python版本

#### C++版本

```
#include <bits/stdc++.h>
using namespace std;
```

```
int main(){
    int max = 0, y, a;
    for(int i = 1;i <= 100;i++){
        cin >> a;
        if(a > max){
            max = a;
            y = i;
        }
    }
    cout << max << endl << y;
    return 0;
}
```

## 三、巩固

### 1.菱形

输入一个奇数  $n$ ，输出一个由 `*` 构成的  $n$  阶实心菱形。

**输入格式**

一个奇数  $n$ 。

**输出格式**

输出一个由 `*` 构成的  $n$  阶实心菱形。

具体格式参照输出样例。

**数据范围**

$1 \leq n \leq 99$

**输入样例：**

5

**输出样例：**

```
  *
 ***
*****
 ***
  *
```

**知识点：**

$n=9$ 时的结果：

涉及的知识点：



1.先获取中心的坐标  $n/2$ ;

2.计算曼哈顿距离, 小于2的点;  $|x-x_c|+|y-y_c| \leq 2$

python版本ok

```
# 打印菱形
# 这里需要注意end的用法, print默认会输出\n,所以把end的标识换成空格就不会 换行
n=9
sum=0
c=9//2
for i in range(0,n+1):
    for j in range(0,n+1):
        if (abs(i-c)+abs(j-c)<=c):
            print("*",end=' ')
        else:
            print(" ",end=' ')
    print('\n',end=' ')
```

输出结果:

```
      *
    * * *
  * * * * *
* * * * * * *
* * * * * * * *
  * * * * * *
    * * * * *
      * * *
        *
```