

第三章 循环结构

第36课 循环嵌套

《信息学奥赛一本通·编程启蒙 C++版》

一、定义

一个循环体内又包含另一个完整的循环结构。内嵌的循环中还可以嵌套循环这就是多层循环。

可以在三个基本循环中，随意选择随意组合。

二、循环方式

假设共两层循环，在外层循环执行一次之后，进入内层循环，内层循环执行完成之后，再进入外层循环……也就是说，一次外层循环对应一次完全的内层循环。

三、嵌套的优化

对于嵌套的内部我们也可以根据题目进行一些优化，比如如果只需要成立时输出一次，我们可以在输出时直接跳出循环可以根据条件利用 `continue` 或者 `break`。

`break` 只能跳出它所在层的循环，如果我们需要跳出两层循环，我们可以借助一个全局 `bool` 类型变量来实现，当满足某个条件的时候，把 `bool` 类型的值改变，先跳出当前层循环；在外层循环再判断 `bool` 值，来确定是否要再跳出一层。

【例 36.1】 计算矩阵边缘元素之和

【题目描述】

输入一个整数矩阵，计算位于矩阵边缘的元素之和。所谓矩阵边缘的元素，就是第一行和最后一行的元素以及第一列和最后一列的元素。

【输入格式】

第一行分别为矩阵的行数 m 和列数 n ($m < 100$, $n < 100$)，两者之间以一个空格分开

接下来输入的 m 行数据中，每行包含 n 个整数（取值范围为 $[0, 9]$ ），整数之间以一个空格分开。

【输出格式】

输出对应矩阵的边缘元素和。

【样例输入】

```
3 3
3 4 1
3 7 1
2 0 1
```

【样例输出】

```
15
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int main(){
4.     int x,y,a,cnt=0;
5.     cin>>x>>y;
6.     for(int i=1;i<=x;i++)
7.         for(int j=1;j<=y;j++){
8.             cin>>a;
9.             if(i==1 || i==x || j==1 || j==y)cnt+=a;
10.        }
11.        cout<<cnt;
12.    return 0;
13. }
```

【例 36.2】 计数问题

【题目描述】

试计算在区间 1 到 n 的所有整数中，数字 $x(0 \leq x \leq 9)$ 共出现了多少次？
例如，在 1 到 11 中，即在 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 中，数字 1 出现了 4 次。

【输入格式】

2 个整数 n , x , 之间用一个空格隔开。对于 100% 的数据， $1 \leq n \leq 1,000,000$, $0 \leq x \leq 9$

【输出格式】

1 个整数，表示 x 出现的次数。

【样例输入】

11 1

【样例输出】

4

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n,x,q=0,sum=0;
4. int main(){
5.     cin>>n>>x;
6.     for(int i=1;i<=n;i++){
7.         q=i;
8.         while(q>0){
9.             if(q%10==x)sum++;
10.            q/=10;
11.        }
12.    }
13.    cout<<sum;
14.    return 0;
15.}
```

【例 36.3】 最大质数

【题目描述】

输入一个正整数 n ，找到 1 到 n 里最大的质数，如果没有质数则输出 "No"。

【输入格式】

输入一个正整数 $n(n \leq 2 \times 10^7)$

【输出格式】

输出 1 到 n 里最大的质数，如果没有质数，则输出 "No"。输出占一行。

【样例输入】

1

【样例输出】

No

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n;
4. bool flag;
5. int main(){
6.     cin>>n;
7.     if(n==1){cout<<"No";return 0;}
8.     if(n==2){cout<<2;return 0;}
9.     for(int i=n;i>=3;i--){
10.         flag=true;
11.         for(int j=2;j*j<=i;j++)
12.             if(i%j==0){
13.                 flag=false;
14.                 break;}
15.         if(flag){
16.             cout<<i;
17.             break;}
18.     }
19.     return 0;
20. }
```

练 36.1 矩阵对角线元素之和

【题目描述】

给定一个 3×3 的矩阵，请你求出对角线元素之和。

【输入格式】

按照行优先顺序输入一个 3×3 矩阵，每个矩阵元素均为整数且取值范围为 $[1, 9]$ 。

【输出格式】

输出从左下角到右上角这条对角线上的元素之和。

【样例输入】

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

【样例输出】

```
13
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int x,sum;
4. int main(){
5.     for(int i=1;i<=3;i++)
6.         for(int j=1;j<=3;j++){
7.             cin>>x;
8.             if(i+j==4) sum+=x;
9.         }
10.    cout<<sum;
11.    return 0;
12.}
```

练 36.2 不定方程求解

【题目描述】

给定正整数 a , b , c 。求不定方程 $ax+by=c$ 关于未知数 x 和 y 的所有非负整数解组数。

【输入格式】

一行，包含三个正整数 a , b , c ，两个整数之间用单个空格隔开。每个数均不大于 1000。

【输出格式】

一个整数，即不定方程的非负整数解组数。

【样例输入】

2 3 18

【样例输出】

4

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int a,b,c,cnt;
4. int main(){
5.     cin>>a>>b>>c;
6.     for(int x=0;x<=c/a;x++)
7.         for(int y=0;y<=c/b;y++)
8.             if(a*x+b*y==c) cnt++;
9.     cout<<cnt;
10.    return 0;
11.}
```

练 36.3 九九乘法表 (趣味编程)

【题目描述】

试编一个程序，输出如图所示的九九乘法表。

$$1*1= 1$$

$$2*1= 2 \quad 2*2= 4$$

$$3*1= 3 \quad 3*2= 6 \quad 3*3= 9$$

$$4*1= 4 \quad 4*2= 8 \quad 4*3=12 \quad 4*4=16$$

$$5*1= 5 \quad 5*2=10 \quad 5*3=15 \quad 5*4=20 \quad 5*5=25$$

$$6*1= 6 \quad 6*2=12 \quad 6*3=18 \quad 6*4=24 \quad 6*5=30 \quad 6*6=36$$

$$7*1= 7 \quad 7*2=14 \quad 7*3=21 \quad 7*4=28 \quad 7*5=35 \quad 7*6=42 \quad 7*7=49$$

$$8*1= 8 \quad 8*2=16 \quad 8*3=24 \quad 8*4=32 \quad 8*5=40 \quad 8*6=48 \quad 8*7=56 \quad 8*8=64$$

$$9*1= 9 \quad 9*2=18 \quad 9*3=27 \quad 9*4=36 \quad 9*5=45 \quad 9*6=54 \quad 9*7=63 \quad 9*8=72 \quad 9*9=81$$

【输入格式】

无。

【输出格式】

如上文所示的乘法表。

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int main()
4. {
5.     int i,j;
6.     for(i=1;i<=9;i++)//控制输出九行
7.     {
8.         for(j=1;j<=i;j++)//控制每行输出 i 个
9.         cout<<i<<"*"<<j<<"="<<setw(2)<<i*j<<" ";
10.        cout<<endl;
11.    }
12.    return 0;
13.}
```

谢谢!

—