第五章 数的存储与组织

第63课 方阵填数

《信息学奥赛一本通·编程启蒙 C++版》

【例 63.1】 蛇形填数

【题目描述】

在 $n \times n$ 方阵里填入 $1, 2, 3, \dots, n \times n$, 要求填成蛇形。例如 $n=4$ 时方阵为:

10 11 12 1

9 16 13 2

8 15 14 3

7 6 5 4

其中, $n \leq 20$ 。

【输入格式】

输入 n 。

【输出格式】

输出题述方阵。 n 行, 每行各数之间用一个空格隔开。

【样例输入】

4

【样例输出】

10 11 12 1

9 16 13 2

8 15 14 3

7 6 5 4

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int dir[4][2]={{1,0},{0,-1},{-1,0},{0,1}}; //方向数组
4. int a[25][25];
5. int main(){
6.     int k = 1,n,fx,fy,d = 0,x,y; //fx, fy 当前焦点位置 x, y 下一个位置 d: 方向
7.     cin >> n;
8.     fx = 1, fy = n;
9.     while(k <= n*n){
10.         a[fx][fy] = k++; //赋值
11.         x = fx + dir[d][0], y = fy + dir[d][1]; //下一个位置
12.         if(a[x][y] != 0 || x > n || x < 1 || y > n || y < 1) //如果新位置上已经有值了或移出了二维数组范围
13.             d = (d + 1) % 4; //变换方向
14.         fx += dir[d][0], fy += dir[d][1]; //更新焦点位置
15.     }
16.     for(int i = 1; i <= n; ++i){
17.         for(int j = 1; j <= n; ++j) cout << a[i][j] << ' ';
18.         cout << endl;
19.     }
20. }
```


【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int a[100][100],n;
4. void f(int x,int y,int n,int c){
5.     if(c<1) return ;
6.     for(int i=0;i<n;i++){
7.         a[x][y+i]=c;
8.         a[x+i][y]=c;
9.         a[x+n-1][y+i]=c;
10.        a[x+i][y+n-1]=c;
11.    }
12.    f(x+1,y+1,n-2,c-1);
13.}
14.int main(){
15.    cin>>n;
16.    f(1,1,2*n+1,n);
17.    for(int i=1;i<=2*n+1;i++){
18.        for(int j=1;j<=2*n+1;j++)printf("%2d",a[i][j]);
19.        printf("\n");
20.    }
21.    return 0;
22. }
```

练 63.1 蛇形三角形

【题目描述】

一个 n 行的蛇形三角形是由 1 开始的自然数依次排列成的一个矩阵上三角形
第一行有 n 个数，接下来每行逐减。具体的规律参考样例。

【输入格式】

输入一个整数 n ， $1 \leq n \leq 100$ 。

【输出格式】

输入 n 行的蛇形三角形

【样例输入】

5

【样例输出】

```
1 3 6 10 15
2 5 9 14
4 8 13
7 12
11
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. long long n,a[105][105],cnt=1;
4. int main(){
5.     cin>>n;
6.     for(int i=1;i<=n;i++){
7.         int x=i,y=1;
8.         while(x>=1){
9.             a[x][y]=cnt;
10.            cnt++;
11.            x--,y++;
12.        }
13.    }
14.    for(int i=1;i<=n;i++){
15.        for(int j=1;j<=n-i+1;j++)cout<<a[i][j]<<' ';
16.        cout<<endl;
17.    }
18.    return 0;
19.}
```

练 63.2 斜行方阵

【题目描述】

输入整数 N，输出相应方阵。

【输入格式】

输入一个整数 n ， $1 \leq n \leq 18$ 。

【输出格式】

一个方阵，每个数字的场宽为 4

【样例输入】

5

【样例输出】

11	7	4	2	1
16	12	8	5	3
20	17	13	9	6
23	21	18	14	10
25	24	22	19	15

【代码实现】

```
1. #include<iostream>
2. using namespace std;
3. int a[20][20],n,x,y;
4. bool check(int x,int y){
5.     return x>=1 && x<=n && y>=1 && y<=n;
6. }
7. int main(){
8.     cin>>n;
9.     int c=0;
10.    for(int i=n;i>=1;i--){
11.        x=1;
12.        y=i;
13.        while(check(x,y)){
14.            c++;
15.            a[x][y]=c;
16.            x++;
17.            y++;
18.        }
19.    }
20.    for(int i=2;i<=n;i++){
21.        x=i;
22.        y=1;
23.        while(check(x,y)){
24.            c++;
25.            a[x][y]=c;
26.            x++;
27.            y++;
28.        }
29.    }
30.    for(int i=1;i<=n;i++){
31.        for(int j=1;j<=n;j++)
32.            printf("%4d",a[i][j]);
33.        printf("\n"); }
34.    return 0;
35.}
```

练 63.3 迂回方阵

【题目描述】

输入整数 N ，输出相应方阵。

【输入格式】

输入一个整数 n ， $1 \leq n \leq 18$ 。

【输出格式】

一个方阵，每个数字的场宽为 4

【样例输入】

5

【样例输出】

17	18	19	20	21
10	11	12	13	22
5	6	7	14	23
2	3	8	15	24
1	4	9	16	25

【代码实现】

```
1. #include<iostream>
2. using namespace std;
3. int a[20][20],n,x,y;
4. bool check(int x,int y){
5.     return x>=1 && x<=n && y>=1 && y<=n;
6. }
7. int main(){
8.     cin>>n;
9.     int c=0;
10.    for(int i=n;i>=1;i--){
11.        x=i;
12.        y=0;
13.        for(int j=1;j<=n-i+1;j++){
14.            c++;
15.            y++;
16.            a[x][y]=c;
17.        }
18.        for(int k=1;k<n-i+1;k++){
19.            c++;
20.            x++;
21.            a[x][y]=c;
22.        }
23.    }
24. }
25. for(int i=1;i<=n;i++){
26.     for(int j=1;j<=n-i+1;j++){
27.         printf("%4d",a[i][j]);
28.         printf("\n"); }
29. return 0;
30. }
```

谢谢!

—