

八皇后问题：使用回溯法，循环里面加个递归

```
#include <stdio.h>
#include <stdlib.h>
#define max 8
int queen[max], sum=0; /* max为棋盘最大坐标 */
void show() /* 输出所有皇后的坐标 */
{
    int i;
    printf("(");
    for(i = 0; i < max; i++)
    {
        printf(" %d", queen[i]);
    }
    printf(")\n");
    sum++;
}

int PLACE(int n) /* 检查当前列能否放置皇后 */
{
    int i;
    for(i = 0; i < n; i++) /* 检查横排和对角线上是否可以放置皇后 */
    {
        if(queen[i] == queen[n] || abs(queen[i] - queen[n]) == (n - i))
        {
            return 1;
        }
    }
    return 0;
}

void NQUEENS(int n) /* 回溯尝试皇后位置,n为横坐标 */
{
    int i;
    for(i = 0; i < max; i++)
```

```

{
    queen[n] = i; /* 将皇后摆到当前循环到的位置 */
    if(!PLACE(n))
    {
        if(n == max - 1)
        {
            show(); /* 如果全部摆好, 则输出所有皇后的坐标 */
        }
        else
        {
            NQUEENS(n + 1); /* 否则继续摆放下一个皇后 */
        }
    }
}
}

int main()
{
    NQUEENS(0); /* 从横坐标为0开始依次尝试 */
    printf("%d", sum);
    system("pause");
    return 0;
}

```

求最大公因子递归

```
public static void main(String[] args) {
    Scanner x=new Scanner(System.in);
    System.out.println("输入两个数");
    int a=x.nextInt();
    int b=x.nextInt();
    int s=a<b?a:b;
    System.out.println("最大公因数是"+jisuan(s,a,b));

}

private static int jisuan(int n,int x,int y) {
    if(n==1||(x%n==0&&y%n==0)){
        return n;
    }
    else{
        return jisuan(n-1,x,y);
    }
}
```