

# 递推的思想

从已知推导未知，从而解决问题。这就是递推的思想

## 递推代码

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

int main(){
    int ad[42],n,i;
    if(n<=10){
        cout << 1 << endl;
    }else{
        ad[1] = 10;
        ad[2] = 10;
        i = 2;
        while(ad[i]<n){
            i+=1;
            ad[i] = ad[i-2]+ad[i-1];
        }
        cout << i << endl;
    }
    return 0;
}
```

递推代码2(方法二)

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

int main(){
    int a,b,c,n,i;
    cin >> n;
    if(n<=10){
        cout << 1 << endl;
    }else{
        a = 10;
        b = 10;
        c = 10;
        i = 2;
        while(c<n){
            i+=1;
            c = a+b;
            a = b;
            b = c;
        }
    }
}
```

```

    cout << i << endl;
}
return 0;
}

```

以上两条代码为教材上P126的广而告之的题（最经典的递推题目）

## 下面的代码为教材P132的例题树荫分区一题（递推思维题）

本题思路：

首先，若干个大树覆盖下会将空地分为多少个区域。再进一步就是计算：再不统计圆外的情况下，若干个椭圆可以将平面划分为多少个区域

划分后得递推式： $n(n+1)+1$

代码如下：

```

#include<iostream>
using namespace std;
const int N = 1001;
int main(){
    int f[N],n,i;
    cin >> n;
    f[1] = 1;
    for(i=2;i<=n;i++){
        f[i] = f[i-1]+2*(i-1);
    }
    cout<<f[n]<<endl;
    return 0;
}

```

方法二

```

#include<iostream>
using namespace std;

int main(){
    int n;
    cin >> n;
    if(n==1){
        cout << 1 << endl;
    }else{
        cout << n*n-n+1<<endl;//或者cout << n*(n+1)+1 << endl;
    }
    return 0;
}

```

}