

4. 栈与队列

调用栈：消除递归

邓俊辉

deng@tsinghua.edu.cn

❖ 动机：递归函数的空间复杂度，主要取决于最大递归深度，而非递归实例总数
为隐式地维护调用栈，需花费额外的处理时间

❖ 方法：显式地维护调用栈，将递归算法改写为迭代版本...

```
❖ int fac( int n ) {  
    int f = 1; //O(1)空间  
    while ( n > 1 )  
        f *= n--;  
    return f;  
}
```

```
❖ int fib( int n ) {  
    int f = 0, g = 1; //O(1)空间  
    while ( 0 < n-- )  
        { g += f; f = g - f; }  
    return f;  
}
```

```
❖ void hailstone( int n ) { //O(1)空间  
    while ( 1 < n )  
        n = n % 2 ? 3*n + 1 : n/2;  
}
```