

## 模型：

find\_sum: 从 0 到 n 枚举，将其中的偶数求和。

```
long long find_sum(long long n)
{
    long long sum = 0;
    for (int i = 0; i <= n; i++)
    {
        if (i % 2 == 0)
            sum += i;
    }
    return sum;
}
```

find\_sum1: 判断 n 的奇偶性，奇数则将 n 减 1，如果为偶数则保留，之后利用等差数列求和公式  $S_n = n * (a_1 + a_n) / 2$  进行计算。

```
long long find_sum1(long long n)
{
    if (n % 2 == 0)
    {
        long long x;
        x = ((n + 1) * (n / 2)) / 2;
        return x;
    }
    else
    {
        long long x;
        x = ((n + 1) * ((n - 1) / 2)) / 2;
        return x;
    }
}
```

## 验证：


在主程序中同时调用 find\_sum 和 find\_sum1, n 从 0 到 100000000 枚举并输出到控制台程序，互相验证正误。（代码如下）

```

1  #include<iostream>
2  #include"find_sum.h"
3  #include"find_sum1.h"
4  using namespace std;
5
6  int main()
7  {
8      long long a;
9      for (a = 0; a <= 1000000000; a++)
10     {
11         cout << find_sum(a) << endl;
12         cout << find_sum1(a) << endl;
13     }
14     return 0;
15 }

```

验证结果：两函数输出结果均正确，且均可复用。（部分结果如下）

 E:\cpp\OOP\1st\x64\Debug\1st.exe

```

275377430
275410620
275410620
275410620
275410620
275443812
275443812
275443812
275443812
275477006
275477006
275477006
275477006
275510800

```