

## Computer Science Problem of the Month

December 2007  
(Solutions due by Dec. 31)

### Background:

In a famous note, "Go To Statement Considered Harmful", Edsger Dijkstra noted that goto statements make it much more difficult to analyze and debug computer programs. Nevertheless, modern programming languages such as C++ still provide the goto statement and will compile and run programs that contain "gotos". This problem illustrates the difficulty of analyzing a program which contains gotos.

### Problem:

The C++ program below uses goto statements to calculate the difference between the sum of the even elements and the sum of the odd elements in an array.

```
int oddMinusEven(int nums[], int n){
    int i=0;
    int sum=0;
    for (;i<n;i++){
        loop1:    if (nums[i] % 2 == 0)
                    goto loop2;
        sum += nums[i];
    }
    for (;i<n;i++){
        loop2:    if (nums[i] %2 !=0)
                    goto loop1;
        sum -= nums[i];
    }
    return sum;
}
```

Analyze the time complexity of this program by finding a formula for the number of steps required to process an array of size  $n$  in the worst case. Your formula should be expressed in terms of  $n$ . You should assume that each operation takes the following number of steps:

<	1 step
==	1 step
%	1 step
++	1 step
goto	1 step
return	1 step
+=	2 steps
-=	2 steps
=	2 steps
[]	2 steps

*All other operations are instantaneous.*