```
// Fragment #5
for( int i = 0; i < n; i++ )
    for( int j = 0; j < i; j++ )
        sum++;

// Fragment #6
for( int i = 0; i < n; i++ )
    for( int j = 0; j < n * n; j++ )
        for( int k = 0; k < j; k++ )
            sum++;
```

5.15. Occasionally, multiplying the sizes of nested loops can give an overestimate for the Big-Oh running time. This happens when an innermost loop is infrequently executed. Repeat Exercise 5.14 for the following program fragment:

```
for( int i = 1; i < n; i++ )
    for( int j = 0; j < i * i; j++ )
        if( j % i == 0 )
            for( int k = 0; k < j; k++ )
                sum++;
```

5.16. In a recent court case, a judge cited a city for contempt and ordered a fine of \$2 for the first day. Each subsequent day, until the city followed the judge's order, the fine was squared (that is, the fine progressed as follows: \$2, \$4, \$16, \$256, \$65536, . . .).

a. What would be the fine on day N?

b. How many days would it take for the fine to reach D dollars (a Big-Oh answer will do)?

5.17. Prove Theorem 5.5. *Hint*: Show that $\sum_2^N \frac{1}{i} < \int_1^N \frac{dx}{x}$. Then show a similar lower bound.

5.18. Construct an example whereby an interpolation search examines every element in the input array.

5.19. Analyze the cost of an average successful search for the binary search algorithm.

In Practice

5.20. Give an efficient algorithm to determine if there exists an integer i such that $A_i = i$ in an array of increasing integers. What is the running time of your algorithm?

5.21. A prime number has no factors besides 1 and itself. Do the following:

a. Write a program to determine if a positive integer N is prime. In terms of N, what is the worst-case running time of your program?

b. Let B equal the number of bits in the binary representation of N. What is the value of B?

c. In terms of B, what is the worst-case running time of your program?