Worksheet Questions

1. If I prove that an algorithm takes $O(n^2)$ worst-case time, is it possible that it takes $O(n)$ on some inputs?

2. If I prove that an algorithm takes $O(n^2)$ worst-case time, is it possible that it takes $O(n)$ on all inputs?

Exercises 10.4

For the code segments in Exercises 7-12, determine which of the orders of magnitude given in this section is the best O to use to express the worst-case computing time as a function of n.

8.
```
// Matrix addition
for (int i = 0; i < n; i++)
   for (int j = 0; j < n; j++)
      c[i][j] = a[i][j] + b[i][j]
```

10.
```
// Bubble sort
for (int i = 0; i < n - 1; i++)
{
   for (int j = 0; j < n - 1; j++)
      if (x[j] > x[j + 1])
      {
         temp = x[j];
         x[j] = x[j+1];
         x[j+1] = temp;
      }
}
```

11.
```
while (n >= 1)
   n /= 2;
```

12.
```
x = 1;
for (int i = 1; i <= n; i++)
{
   for (int j = 1; j <= x; j++)
      cout << j << endl;
   x *= 2
}
```