

EXERCISES FOR SECTION 2.4

SELF-CHECK

1. Determine how many times the output statement is displayed in each of the following fragments. Indicate whether the algorithm is $O(n)$ or $O(n^2)$.

a.

```
for (int i = 0; i < n; i++)
    for (int j = 0; j < n; j++)
        System.out.println(i + " " + j);
```

b.

```
for (int i = 0; i < n; i++)
    for (int j = 0; j < 2; j++)
        System.out.println(i + " " + j);
```

c.

```
for (int i = 0; i < n; i++)
    for (int j = n - 1; j >= i; j--)
        System.out.println(i + " " + j);
```

d.

```
for (int i = 1; i < n; i++)
    for (int j = 0; j < i; j++)
        if (j % i == 0)
            System.out.println(i + " " + j);
```

- a. $O(N^2)$: the program will run through n^2 amount of times since the first for loop runs until the end of n and the nested loop also runs to n .
- b. $O(N)$: the program will run through n times on the first loop and an addition 2 times for each n value. The program will run through $N + 2$ times each operation
- c. $O(N^2)$: the program will run through n times for the first for loop and the nested for loop will run from the current index i up to $n-1$ for each operation (M). $M < N$ but for larger data types the amount less will be less significant so it may be a bit less than n^2 but closer to n^2 than to n .
- d. $O(N^2)$: the program will run on the first for loop n times, the nested for loop will run up to the current index i . For larger data the program will run up to the current index i which will be significantly more than $O(N)$ but still less than $O(N^2)$, still closer to $O(N^2)$ due to the nested loop counting up to the current index which will reach the value of N eventually.