Section 02: Asymptotic Analysis

Finding bounds

For each of the following code blocks, construct a mathematical function modeling the worst-case runtime of the code in terms of \( n \). Then, give a tight \( \text{big-O} \) bound of your model.

(a)  
```java
int x = 2;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= 4*i; j++) {
            x = x*x;
        }
    }
```

(b)  
```java
int x = 0;
    for (int i = 0; i < n; i++) {
        if (i % 5 == 0) {
            for (int j = 0; j < n; j++) {
                if (i == j) {
                    x += i * j;
                }
            }
        }
    }
```

Recurrence Relations

(a) What is the runtime of the following function as a function of \( n \).
```java
int Compute(int n){
    if(n<=2){
        return 1;
    }
    else{
        n = Math.sqrt(n); // sqrt(n) computes the square root of n
        return Compute(n);
    }
}
```

Hint: First, write down the recurrence relation for \( T(n) \), and solve the recurrence relation.

(b) Solve the following recurrence relation: 
\[ T(n) = \begin{cases} 
1 & \text{if } n = 1 \\
8T(n/2) + n^2 & \text{otherwise} 
\end{cases} \]