Homework 2

Computer Science 211
Fall 2019 Due Thursday, September 19

1. Order the following functions by growth rate:

\[ 4n \log n + 2n \quad 2^{10} \quad 2^{\log n} \]
\[ 3n + 100 \log n \quad 4n \quad 2^n \]
\[ n^2 + 10n \quad n^3 \quad n \log n \]

2. What is the big-oh running time of this code?

```java
public static int example1 (int[] a) {
    int n = a.length;
    int sum = 0;
    for (int j=0; j<n; ++j)
        sum += a[j];
    return sum;
}
```

3. What is the big-oh running time of this code?

```java
public static int example2 (int[] a) {
    int n = a.length;
    int sum = 0;
    for (int j=0; j<n; j += 2)
        sum += a[j];
    return sum;
}
```

4. What is the big-oh running time of this code?

```java
public static int example3 (int[] a) {
    int n = a.length;
    int sum = 0;
    for (int j=0; j<n; ++j)
        for (int k=0; k<=j; ++k)
            sum += a[j];
    return sum;
}
```
5. What is the big-oh running time of this code?

```java
public static int example4 (int[] a) {
    int n = a.length;
    int prefix = 0;
    int sum = 0;
    for (int j=0; j<n; ++j) {
        prefix += a[j];
        sum += prefix;
    }
    return sum;
}
```

6. What is the big-oh running time of this code?

```java
public static int example5 (int[] a, int[] b) {
    // we assume that the arrays have the same length
    int n = a.length;
    int count = 0;
    for (int i=0; i<n; ++i) {
        int sum = 0;
        for (int j=0; j<n; ++j) {
            for (int k=0; k<=j; ++k)
                sum += a[j];
            if (b[i] == sum) count++;
        }
    }
    return count;
}
```