Name: Michael Buzzetta Date: September 21, 2023

Pledge: I pledge my Honor that I have abided by the Stevens honor system

Give the asymptotic complexity of the following functions. Choose the most appropriate notation from among O, θ, and Ω. Give only a single answer for each question (giving more than one answer will result in a zero for that question).

1.
```
void function1(int n) {
    for (int i = 1; i <= n; i++) {
        for (int j = i; j <= n; j += 2) {
            cout << "*";
        }
    }
}
```
Answer: **θ(n^2)**

2.
```
void function2(int n) {
    int count = 0;
    for (int i = 1; i * i * i <= n; i++) {
        count++;
    }
    cout << count;
}
```
Answer: **θ(n^1/3)**

3.
```
void function3(int n) {
    int count = 0;
    for (int i = 1; i * i <= n; i++) {
        for (int j = 1; j + n/2 <= n; j++) {
            for (int k = 1; k <= n; k *= 2) {
                count++;
            }
        }
    }
    cout << count;
}
```
Answer: **θ ((n^(3/2))(log 2 (n))/2)**

4.
```
void function4(int n) {
    int count = 0;
    for (int i = n/2; i <= n; i++) {
        for (int j = 1; j <= n; j *= 2) {
            for (int k = 1; k <= n; k *= 2) {
                count++;
            }
        }
    }
    cout << count;
}
```
Answer: $\boldsymbol{\theta(n\log(2)^2(n))}$

5.
```
void function5(int n) {
    if (n % 2 == 0) {
        return;
    }
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            cout << "*";
            break;
        }
    }
}
```
Answer: **$\Omega(n)$**

6.
```
void function6(int n) {
    int count = 0;
    for (int i = 1; i <= n/2; i++) {
        for (int j = 1; j <= n/3; j++) {
            for (int k = 1; k <= n/4; k++) {
                count++;
            }
        }
    }
    cout << count;
}
```
Answer: $\boldsymbol{\theta(n^3)}$

7.
```
void function7(int n) {
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j += i) {
            cout << "*";
        }
    }
}
```
Answer: $\boldsymbol{\theta(nlog(n))}$

8.
```
void function8(int n) {
    int i = 1, s = 1;
    while (s <= n) {
        i++;
        s += i;
        cout << "*";
    }
}
```
Answer: $\boldsymbol{\theta(sqrt(n))}$

9. Processing Arrays
    a. Suppose you have an unsorted array of integers of length n and want to sum all the elements inside it. What is the running time of your algorithm? **$\theta(n)$**

b. Suppose you have an unsorted array of integers of length n and want to determine if all the values inside are positive. What is the running time of your algorithm? **O(n)**
c. Suppose you have a sorted array of integers of length n and want to determine the median value. What is the running time of your algorithm? **θ (1)**

10. True or False: $f(n) = 5n^2 + 4n + 8 \ \in \ \theta(n^2)$ Answer: **True**
Then, if true, prove it by giving integer values for the required constants c_1, c_2, and n_0. Choose the tightest values possible for the c_1 and c_2 constants. If false, show a contradiction.
5n^2 <= 5n^2+4n+8 <=6n^2 for all n >=6