1. **(10 points)** Order the following functions by growth rate: \( N, \sqrt{N}, N^{1.5}, N^2, N \log N, N \log \log N, N \log^2 N, N \log(N^2), 2/N, 2^N, 2^{N/2}, 37, N^2 \log N, N^3 \). Indicate which functions grow at the same rate.

2. **(10 points)** Compute the running time \( T(n) \) of the program fragment below and provide an analysis of the running time (Big-Oh notation will do). For convenience, assume that operations inside for loops take constant time, i.e. \( \Theta(1) \).

```plaintext
sum = 0;
for (i = 0; i < n; i++)
    for (j = 0; j < i * i; j++)
        for (k = 0; k < j; k++)
            sum++;
```

3. **(10 points)** An algorithm takes 0.5\( ms \) for input size 100. How long will it take for input size 500 if the running time is the following (assume low-order terms are negligible): linear, \( O(n \log n) \), quadratic, and cubic?

4. **(10 points)** Give an \( O(\log N) \) algorithm to determine if there exists an integer \( i \) such that \( A_i = i \) in an array of integers \( A_1 < A_2 < A_3 < \ldots < A_N \). For example, in \{−10, −3, 3, 5, 7\}, \( A_3 = 3 \). In \{2, 3, 4, 5, 6, 7\}, there is no such \( i \).

5. **(10 points)** Show that \( X^{62} \) can be computed with only eight multiplications.

6. **(10 points)** Given two functions \( T_1(n) = 20n \) and \( T_2(n) = n^2 + 10n - 200 \) and assuming integer \( n > 0 \), what is the minimum value of \( n \) such that \( T_1(n) < T_2(n) \)?

7. **(10 points)** Is a linear-time algorithm always faster than a quadratic-time algorithm? Why or why not?