

Answer all questions. No question should take more than one page.

1. Give formal definitions for O, Omega, Theta, P, and NP.
2. Sketch pseudocode for either Prim's algorithm, or Kruskal's algorithm (specify which one you are doing). Also include a small example problem, to illustrate how the algorithm works.
3. What is the Big-O complexity for the following function?

```
int f(int n)
{
    int i, j, k;

    k = 0;
    for (i = 0; i < n; i = i + 1)
        for (j = 0; j < 1000; j = j + 1)
            k = k + i * j;

    return k;
}
```

4. Use dynamic programming to find the longest common subsequence of APPLE and PLATE (the longest common subsequence is the longest series of letters that appear in the same order in both words). Show the matrix for your work.
5. Describe how you might use a polynomial time algorithm for the Traveling Salesman Problem (if one were to exist) to solve the Hamiltonian Cycle problem.