Name:________________________________________________________________________

The entire examination is 280 points.

1. True or False. [5 points each]

(a) _____ Using path compression and making sure to merge the small tree with the large tree, $n$ operations for union/find may be done in $O(n)$ time, given that you start with $n$ singleton trees.

(b) _____ If a planar graph $G$ (a graph is called planar if it can be embedded in a plane with no edge crossings) has $n$ vertices, then $G$ has $O(n)$ edges.

(c) _____ Kruskal’s algorithm is a greedy algorithm.

(d) _____ Greedy algorithms are used because they’re quick and easy to write, but they’re never optimal.

(e) _____ If a hash table has size $2n$ but holds only $n$ items, and if the hash function is pseudo-random, then, with very high probability, there will be no collision.

2. Fill in the blanks (5 points each blank).

(a) A graph with 60 nodes has no more than _________ edges.

(b) An acyclic connected graph with $n$ nodes has _________ edges.

(c) A strongly connected directed graph with $n$ nodes must have at least _________ edges.

(d) If a hash function is used to build a search structure, not a perfect hash table, what properties should it have? (I want three properties.)
3. Give the best possible asymptotic time complexity of each of these code fragments.

(a) [10 points]
```cpp
int m = n*n;
while (m > 0)
{
    cout << "hello world" << endl;
    m = m/2;
}
```

(b) [10 points]
```cpp
int m = n;
while (m > 0)
{
    for ( int i = 0; i < m ; i++ )
        cout << "hello world" << endl;
    m = m/2;
}
```
4. [20 points] Write an appropriate loop invariant for the inner loop in this function.

```cpp
void bubblesort(vector<int> & x)
{
    for(int i = 0; i < x.size; i++)
        for(int j = x.size-1; j > i; j--)
            if(x[j] < x[j-1])
                {
                    int temp = x[j];
                    x[j] = x[j-1];
                    x[j-1] = temp;
                }
}
```

5. [20 points] Write a complete C++ program that:

   (a) prompts the user to enter three integers,
   (b) reads three integers from the keyboard,
   (c) writes the largest of those three numbers to the screen.
6. [30 points]
Consider the weighted directed graph represented by the matrix below. A blank entry indicates no edge.

(a) Write the nodes in a topological order.

(b) Solve the single source minimum weight path problem for this graph, with start node A. Your answer should consist of two arrays: minimum weights and back pointers.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>L</th>
<th>M</th>
<th>O</th>
<th>R</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>-1</td>
<td>7</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
<td>7</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>-3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>2</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O</td>
<td>4</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R</td>
<td></td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
7. [25 points] Explain how you would use a search structure to implement a sparse array. The space below should be sufficient, but you can write on the back of this page if necessary.
8. [30 points] In FORTRAN, all matrices (i.e., arrays) are stored in column-major order, and indices always start at 1 (not 0, as with C++). A FORTRAN program contains a declaration for a $10 \times 8 \times 20$ 3-dimensional matrix of type FLOAT, called $A$. Each variable of type FLOAT uses two words (address locations).

The compiler allocates a block of space, starting with word 1025, for $A$. Where will the variable $A(5, 4, 16)$ be stored? (FORTRAN uses parentheses instead of brackets to indicate array indices.)
9. [20 points] Explain when you could use the Floyd-Warshall algorithm, and give pseudo-code.
10. [30 points] Walk through the steps of Dijkstra’s algorithm to solve the single source minimum path problem for the graph shown below, where $s$ is the start node.

11. [30 points] Some problem (or problems) on dynamic programming.