Instructions: You have 2 hours and 50 minutes to complete this exam. The exam is closed book and closed notes, with the exception of one 8.5” x 11” sheet of paper. No calculators are allowed. Make sure to show your work on all problems. No credit will be given for answers without sufficient work.

Problem 1

Problem 2

Problem 3

Problem 4

Problem 5

Problem 6

Problem 7

Problem 8

TOTAL
1) 12 points
Use proof by induction to prove that \((n+1)^2 < 2n^2\), for all \(n \geq 3\).
2) 13 points

Write C-like or Java-like pseudocode to implement a recursive sorting function that operates in the following way. When given an array of n integers to sort, the function ignores the first element of the array, recursively sorts the remaining n-1 elements, and then goes through the array to place the first element in its proper place shifting the other elements to the left as appropriate.
3) 12 points

a) How many 3-of-a-kind poker hands are there? A poker hand consists of 5 cards. A 3-of-a-kind hand has 3 cards of the same rank. Make sure not to count 4-of-a-kind hands and full-house hands. A full house is made up of a 3-of-a-kind and a pair together.

b) What is the probability of being dealt a 3-of-a-kind hand when 5 cards are dealt from a 52-card deck? Assume that all hands are equally likely to be dealt.
4) 13 points

A complete ternary tree is a tree where every non-leaf node has exactly 3 children and all leaves are at the same depth. Prove that a complete ternary tree of height h has exactly \((3^{h+1} - 1)/2\) nodes.
5) 12 points
   a) Draw the heap that results after the following items are inserted in sequence into an empty heap: 36, 15, 9, 24, 17, 21, 55, 12, 34. Assume the larger the value of the item, the higher is its priority.

   b) Starting from your answer to a), show the resulting heap after 3 items are removed
6) 13 points

Use Prim’s Algorithm and Dijkstra’s Algorithm to find a minimum spanning tree and shortest path tree for the below graph. For Dijkstra’s Algorithm, use \( v_6 \) as the source node.
7) 12 points

Reduce the following state machine to a minimum number of states:

<table>
<thead>
<tr>
<th>Current State</th>
<th>Next State</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>X = 0</td>
<td>X = 1</td>
</tr>
<tr>
<td></td>
<td>X = 0</td>
<td>X = 1</td>
</tr>
<tr>
<td>A</td>
<td>H</td>
<td>C</td>
</tr>
<tr>
<td>B</td>
<td>C</td>
<td>D</td>
</tr>
<tr>
<td>C</td>
<td>H</td>
<td>B</td>
</tr>
<tr>
<td>D</td>
<td>F</td>
<td>H</td>
</tr>
<tr>
<td>E</td>
<td>C</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>G</td>
</tr>
<tr>
<td>G</td>
<td>G</td>
<td>C</td>
</tr>
<tr>
<td>H</td>
<td>A</td>
<td>C</td>
</tr>
</tbody>
</table>
8) 13 points

Draw a parse tree for the expression “w c w c { s ; s ; w c s ; }” with the following context-free grammar:

\[
\begin{align*}
    \langle S \rangle & \rightarrow w \ c \ \langle S \rangle \\
    \langle S \rangle & \rightarrow \{ \ \langle L \rangle \} \\
    \langle S \rangle & \rightarrow s ; \\
    \langle L \rangle & \rightarrow \langle L \rangle \ \langle S \rangle \\
    \langle L \rangle & \rightarrow \epsilon
\end{align*}
\]