1(a) Formally prove the function \( f(n) = 2n^3 - 10n^2 - 100n + 5 \) is \( \Theta(n^3) \).

<table>
<thead>
<tr>
<th>GRADE</th>
<th>/20</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>SUM</td>
<td>/100</td>
</tr>
</tbody>
</table>

1(b) The following “proof by induction” attempts to prove that “all horses in universe are of the same color”! Obviously, there must be something wrong with the proof. So, find exactly what is wrong with the proof. Circle the wrong step(s) and explain why the steps are wrong.

**Claim:** Every set of \( n \) horses are of the same color.

**Proof:** By induction on \( n \).

1. (Induction Base) For \( n = 1 \), the claim is obviously correct.

2. (Induction Hypothesis) For any \( n \geq 2 \), assume every set of \( n - 1 \) horses are of the same color.

3. Consider a set of \( n \) horses. Number the horses \( 1 \ldots n \).

4. By the induction hypothesis, horses \( 1 \ldots n - 1 \) have the same color.

5. By the induction hypothesis, horses \( 2 \ldots n \) also have the same color.

6. There is at least one common horse between the two subsets \( 1 \ldots n - 1 \) and \( 2 \ldots n \).

7. Therefore, all \( n \) horses are of the same color.
2(a) Find the exact solution of the following recurrence. Assume \( n \) is a power of 2.

\[
T(n) = \begin{cases} 
8T(n/2) + n^2, & n > 1 \\
1, & n = 1.
\end{cases}
\]

2(b) Prove by induction that the following recurrence has the solution \( T(k) = Ak^2 + Bk + C \) and find the constants \( A, B, C \).

\[
T(k) = \begin{cases} 
T(k - 1) + k, & k > 1 \\
1, & k = 1.
\end{cases}
\]
3(a) Write a recursive C or C++ function to compute the **height** of every node in a binary tree. (We define the height of a leaf node to be 0.) Assume that each node has LCHILD and RCHILD pointers and a field for storing the height.

3(b) Analyze the time complexity of the above algorithm in terms of the number of nodes $n$ and the height of the tree $h$. 
4(a) Insert element 10 in the following heap.
(First complete the picture by drawing lines from each node to its children.)

```
    8
   /   \
 15    12
 /     / \
20    15 14 13
/      \
23  21   15 17 15 14 13
```

What is the time complexity of one insert operation for a heap of $n$ elements? Explain briefly.

4(b) Do a Delete-Min operation on the following heap.

```
    8
   /   \
 15    12
 /     / \
20    15 12 13
/      \
23  21   15 17 14 13
```

What is the time complexity of one Delete-Min operation for a heap of $n$ elements? Explain briefly.
5(a) Insert element 58 in the following AVL tree. (First complete the picture by drawing lines from each node to its children.) Show the result after the initial insert, and then after each rotate operation needed to rebalance.

```
  50
 /   \
40   65
 / \
35 55 70
  |
52 60 75
```

5(b) Delete element 35 from the following AVL tree. Show the result after the initial delete, and after each rotate operation needed to rebalance.

```
  50
 /   \
40   65
 / \
35 55 70
  |
52 60 75
```