A series of if-then-else constructs can be modeled with an $m$-ary decision tree where $m$ is equal to _____.

A balance scale can be modeled with an $m$-ary decision tree where $m$ is equal to _____.

In a collection of 10 coins, 3 coins are counterfeit and weigh more than the genuine coins. Find a good lower bound on the number of balance scale weighings needed to identify all the fake coins. _____

A) $10^3$  B) $3^{10}$  C) $\log_2 C(10,3)$  D) $\log_2 P(10,3)$  E) $\log_3 C(10,3)$  F) $\log_3 P(10,3)$  G) $\log_2 10^3$  H) $\log_3 10^3$

According to the Lower Bound Theorem, the best (most efficient) worst case complexity for comparison-based search is _____.

A) $O(1)$  B) $O(\log_2 n)$  C) $O(n)$  D) $O(n \log_2 n)$  E) $O(n^2)$  F) $O(n^3)$  G) $O(2^n)$  H) $O(n!)$

According to the Lower Bound Theorem, the best (most efficient) worst case complexity for comparison-based sort is _____.

A) $O(1)$  B) $O(\log_2 n)$  C) $O(n)$  D) $O(n \log_2 n)$  E) $O(n^2)$  F) $O(n^3)$  G) $O(2^n)$  H) $O(n!)$

If you can solve one of these problems in polynomial time, then all problems in this classification can be solved in polynomial time. _____.

A) $P$  B) $NP$  C) $NP-Hard$  D) $NP-Complete$  E) $NP-Easy$  F) $NP-Equivalent$  G) $P \neq NP$

Induction is natural to use to prove correctness of __________________ algorithms while loop invariants can be used to prove correctness of __________________ algorithms.

The following is an implementation of an algorithm to recursively calculate the $n$th triangular number:

```c
unsigned int triNum( unsigned int n )
{
    if ( n == 1 )
    {
        return 1;
    }
    else
    {
        return n + triNum( n - 1 );
    }
}
```

The closed-form solution to this is $\frac{n(n+1)}{2}$

What is the recursive part of the recurrence relation for this algorithm? $T(n) = \text{_______________________________}$

What is the run time complexity of this algorithm? ________

What gets calculated when you add triNum(n) + triNum(n-1)? ____________________________
In the notes, part of the recursive binary search was wrong. Here is a copy from the notes with line numbers:

```c
1 int binarySearch( int key, int data[], int low, int high )
2    { if ( high < low )
3      return -1;
4    }
5    int midPoint = low + (high - low) / 2;
6    if ( key == data[midPoint] )
7      return midPoint;
8    else
9      { if ( key < data[midPoint] )
10         return binarySearch( key, data, low, midPoint - 1 );
11      }
12      else
13         return binarySearch( key, data, low + 1, high );
14 } // end of binarySearch()
```

State the line number where the error occurs (as discussed in class). _____

And then rewrite the line to fix the code.

In the back story behind the Towers of Hanoi problem discussed in class, how many rings were the monks tasked with? _____

Which language is Niklaus Wirth not known for? _____

A) Algol-W    B) Pascal    C) Ada    D) Modula    E) Oberon    F) Euler    G) PL/0

Given the initial order of ints in an array as: 5, 7, 9, 1, 2 what is the order of the elements after 2 iterations of the selection sort algorithm covered in class and one of the HW exercises? _____ _____ _____ _____ _____

What is the run time complexity of the selection sort? _____

A) O(1)    B) O(log_2 n)    C) O(n)    D) O(n log_2 n)    E) O(n^2)    F) O(n^3)    G) O(2^n)    H) O(n!)

Which finite state automaton correctly recognizes words of the language \((ab)^n\) for \(n \geq 1\)? _____

\(v_0\) is the start node. \(w\) is a terminal node. A node labeled with both \(v_0\) and \(w\) is both a start and terminal node.

A)                  B)                  

C)                  D)