Sample Questions
Csci 1112
A. Bellaachia

- **Important Series :**
  - $S(N) = 1 + 2 + \ldots + N = \sum_{i=1}^{N} i = N(1+N)/2$
  - **Sum of squares:**
    - $\sum_{i=1}^{N} i^2 = \frac{N(N+1)(2N+1)}{6} \approx \frac{N^3}{3}$ for large N
  - **Sum of exponents:**
    - $\sum_{i=1}^{N} i^k \approx \frac{N^{k+1}}{|k+1|}$ for large N and k ≠ -1
  - **Geometric series:**
    - $\sum_{i=0}^{N} A^i = \frac{A^{N+1}-1}{A-1}$
    - Special case when A = 2
      - $2^0 + 2^1 + 2^2 + \ldots + 2^N = 2^{N+1} - 1$
  - Some Math Formulas
    - Properties of logarithms:

      $\log_b(xy) = \log_b x + \log_b y$
      $\log_b (x/y) = \log_b x - \log_b y$
      $\log_b xa = a\log_b x$
      $\log_b a = \log_x a / \log_x b$

    - Properties of exponentials:

      $a^{(b+c)} = a^b a^c$
      $a^{bc} = (a^b)^c$
      $a^b / a^c = a^{(b-c)}$
      $b = a^{\log_a b}$
      $b^c = a^{c*\log_a b}$

1. **Problem**

```
For i=1 to n
        m=i, j=1;
        While (j<=m) do
                O(1);
                j=j+1;
        endwhile;
```

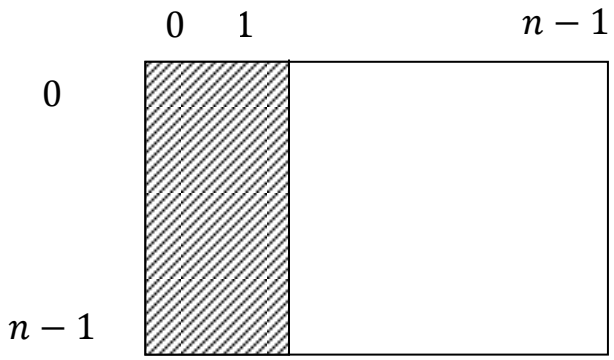
```
j=1;
While (j<=n) do
                for i=1 to j do
                O(1);
        endfor;
        j = j*4;
endwhile;
```

Show that the time complexity of the following code is: $O(n^2 \log n)$

```
i = 1;
while( i < n ){
        doIt();
        i = i*3;
}
```

2. **Problem**

The following M[n][n] matrix is considered a space matrix.



Example:

```
11000
11000
11000
11000
11000
```

(1) Characterize the non zero elements of M.
(2) Give a mapping function required to store M in a one-dimensional array M'.

3. **Problem**

(1) Given two sorted singly linked list L1 and L2, write a pseudo-code that merges the two list in a new list L
(2) Repeat (1) without using a new list L; merge L1 and L2 in L1.

4. **Problem**
   Write a non recursive pseudo-code that prints the elements of a singly linked list in reverse, using only constant extra space. What is the time complexity of your algorithm?

5. **Problem**
   Write a pseudo-code to swap two adjacent nodes by adjusting only the links(and not the data) using:
   a. Singly linked list
   b. Doubly linked list

6. **Problem**
   In this problem we would like to delete the first inserted k elements of a stack. The following example shows the state of the stack after we delete the first inserted three elements in a stack:

   a) Write an algorithm that deletes the first **k** inserted elements of a stack S.
   b) What is the time complexity of your algorithm?

7. **Problem**
   Write a function to test if a string is a palindrome using a stack. What is the time complexity of your algorithm?

8. **Problem**
   Given an expression string, write a program using a stack that checks whether the pairs and the orders of "{","}","(",")","[","]" are correct in the string. For example, the program should print true for the following expression:

   "[(a+b)]-{a-b-c}+ {[(e+f)+(t+h)]-(a-g)}"

   And false for this expression: "[(a+b]-c)"

9. **Problem**
   Write a non recursive pseudo-code that prints the elements of a singly linked list using a stack data structure. What is the time complexity of your algorithm?

10. Problem

Given the following code:

```
Stack Mysterystack = new Stack(10);
int N = 50;
while (N >1 ) {
      Mysterystack.push(N % 2);
   N = N / 2;
}
System.out.println("==============================");
while (!Mysterystack.isEmpty()){
    System.out.println(Mysterystack.top());
    Mysterystack.pop();
}
```

(a) Explain in general what does this code return?

(b) What is the output of this code for the following?

- N = 30 [Ans: 1110]
- N = 40 [Ans: 01000]
- N = 60 [Ans: 11100]

11. Problem

Write a pseudo-code that determine whether a string is in the language L, where

$$L = \{s \in \{a,b\}^* \; such that \; s \; is \; the \; form \; a^n b^n\}$$

12. Problem

Given two queues Q1 and Q2, write a pseudo-code that counts how many elements in the same position of the two queues are equal. The queues may not be the same length. Do not modify the queues.

13. Problem:

Consider the following recursion function:

```
void function(int n)
{
      cout << n << endl;
      if (n>1)
            function(n-1);
}
```

(1) What is the output of this function when n = 5?

(2) What is the output of this function when n = 8?

(3) Explain what does this function do?

14. Problem

Give a recursive method for removing all the elements from a stack.

15. Problem

The nth power of a non-zero number can be recursively defined as follows:

$$x^n = \begin{cases} x * x^{n-1} & n > 0 \\ 0 & n = 0 \end{cases}$$

Write a recursive function that computes the nth power of a number using the recurrence relation above.

16. Problem

The nth power of a non zero number can also be recursively defined as follows:

$$x^n = \begin{cases} x^{\frac{n}{2}} * x^{\frac{n}{2}} * x & n\ is\ odd\ and\ \ n\ > 0 \\ x^{\frac{n}{2}} * x^{\frac{n}{2}} & n\ is\ even\ and\ \ n\ > 0 \\ 0 & n = 0 \end{cases}$$

Write a recursive function that computes the nth power of a number using the recurrence relation above.

17. Problem

Write an algorithm that returns the minimum value in a binary search tree. (Hint: Go to the left child of the root and keep going to the left child until you come to a leaf node. This node is the minimum.) What is time complexity of your algorithm?

18. Problem

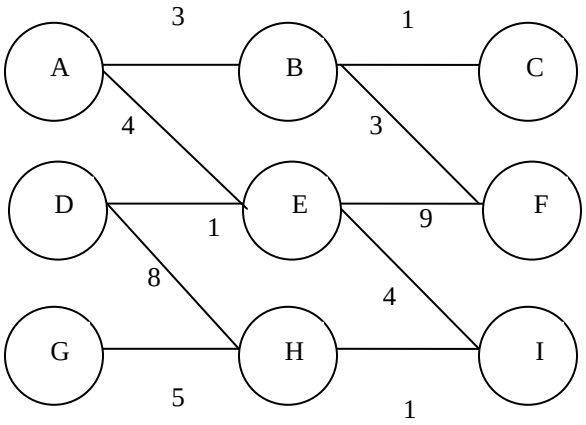
Write an algorithm that returns the maximum value in a binary search tree. (Hint: Go to the right child of the root and keep going to the right child until you come to a leaf node. This node is the maximum.) What is time complexity of your algorithm?

19. Problem

Write a recursive algorithm that returns the total number of nodes in a binary tree. What is time complexity of your algorithm?

20. Problem

Given the following graph



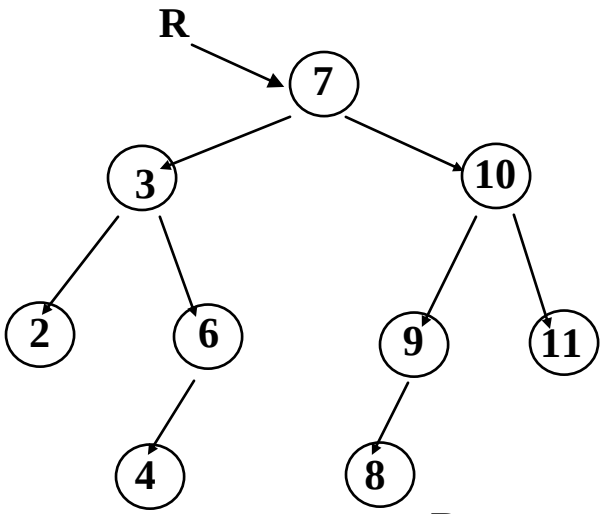
(1) Give two different DFS traversals for G and their corresponding DFS spanning trees
(2) Give two different BFS traversals for G and their corresponding DFS spanning trees

21. Problem

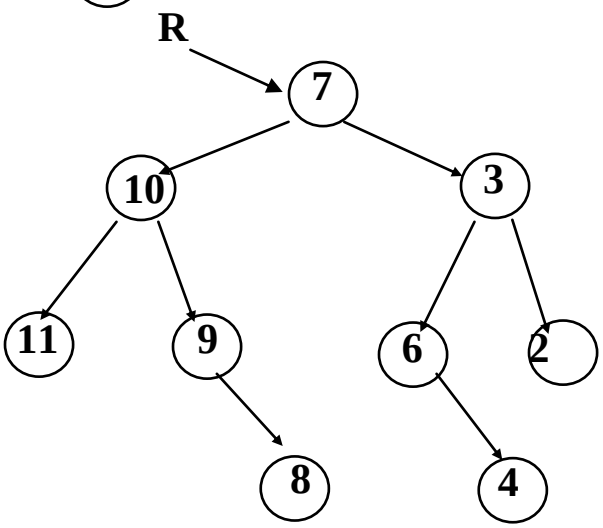
Write a program that returns the maximum degree of an undirected graph G. What is the time complexity of your program? Explain.

22. Problem

Write an algorithm that takes a binary tree and return a new tree. The new tree swaps the left and right child of every node. For example:



The swapped tree is:



23. Problem

Write an algorithm that copies a binary tree T1 into a new tree T2.

24. Problem

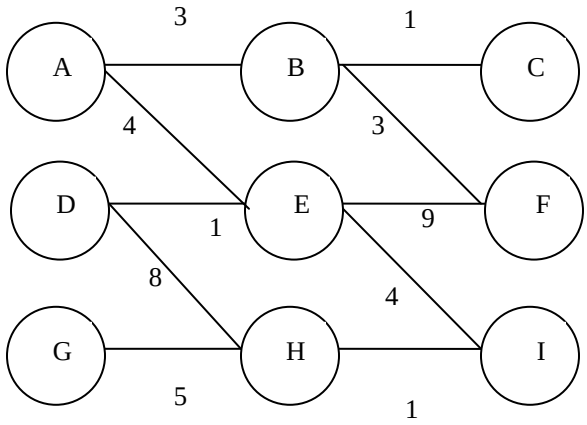
(1) Show the result of inserting 3, 1, 4, 6, 9, 2, 5, 7, 0 into an initially empty binary search tree B. Draw you tree.

(2) Delete the root and draw your BST B'.

(3) Delete the node that contains 6 in B' and draw your BST B''.

25. Problem

We would like to change Breadth First search (BFS) traversal to a new traversal called SBFS. Instead of using a queue, SBFS uses a stack. Give a DFS, BFS and SBFS spanning trees for the following graph:



26. Problem

(1) Show the result of inserting 3, 1, 4, 6, 9, 2, 5, 7, 0 into an initially empty min-heap B. Draw your heap.
(2) Delete the root of B and draw the new heap B'.
(3) Delete the root of B' and draw the new heap B''.

27. Problem

Every character in a given computer is decoded as two digits, according to the following code:

0-9	00 through 09
A-I	11 through 19
J-R	21 through 29
S-Z	32 through 39

Now consider this hashing-coding problem. There is a hash table of 100 entries, indexed by two digits decimal subscripts. We would like to hash words of 4 letters. A hashing method is proposed in which h(w) is computed by dividing the word by 10 (integer division), then taking the rightmost two digits of the result. Find h(w) for each of the keys MARY, JACK, WELL, MACK, BOLD.

28. Problem

Consider inserting the keys 10, 22, 31, 4, 15, 28, 17, 50, 51 into a hash table of length m = 13 using open addressing with the auxiliary hash function F(x) = x mod 13. Show the hash table of the given list using the following:

- Linear probing,
- Quadratic probing

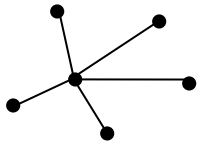
- Double hashing with F1(x) = x and F2(x) = 1 + (x mod(m-1)).

29. Problem

Give an algorithm that checks whether an undirected graph G is a complete graph. State the data structure you are using to store your graph.

30. Problem

An example of an undirected star graph is:



Give a recursive strategy to check whether a graph S is a star graph.