

BSC. Information Technology

Class:SYIT

SEM III

Data Structure

Sample Questions

Multiple Choice Questions

- 1) Which of the following statements for a simple graph is correct?
 - a. Every path is a trail
 - b. Every trail is a path
 - c. Every trail is a path as well as every path is a trail
 - d. Path and trail have no relation
- 2) Which of the following properties does a simple graph not hold?
 - a. Must be connected
 - b. Must be unweighted
 - c. Must have no loops or multiple edges
 - d. Must have no multiple edges
- 3) A graph with all vertices having equal degree is known as a _____
 - a. Multi Graph
 - b. Regular Graph
 - c. Simple Graph
 - d. Complete Graph
- 4) A connected planar graph having 6 vertices, 7 edges contains _____ regions.
 - a. 15
 - b. 3
 - c. 1
 - d. 11
- 5) What is the number of edges present in a complete graph having n vertices?
 - a. $(n*(n+1))/2$
 - b. $(n*(n-1))/2$
 - c. n
 - d. Information given is insufficient
- 6) A connected planar graph having 6 vertices, 7 edges contains _____ regions.
 - a. 15

- b. 3
 - c. 1
 - d. 11
- 7) What is the maximum number of edges in a bipartite graph having 10 vertices?
- a. 24
 - b. 21
 - c. 25
 - d. 16
- 8) For a given graph G having v vertices and e edges which is connected and has no cycles, which of the following statements is true?
- a. $v=e$
 - b. $v = e+1$
 - c. $v + 1 = e$
 - d. $v = e-1$
- 9) A vertex of in-degree zero in a directed graph is called a/an
- a. Root vertex
 - b. Isolated vertex
 - c. Sink
 - d. Articulation point
- 10) A binary tree in which all its levels except the last, have maximum numbers of nodes, and all the nodes in the last level have only one child it will be its left child. Name the tree.
- a. Threaded tree
 - b. Complete binary tree
 - c. M-way search tree
 - d. Full binary tree
- 11) The number of edges in a complete graph of n vertices is
- a. $n(n+1)/2$
 - b. $n(n-1)/2$
 - c. $n^2/2$
 - d. n
- 12) An adjacency matrix represent of a graph cannot contain information of:
- a. Nodes

- b. Edges
 - c. Direction of edges
 - d. Parallel edges
- 13) Consider an undirected graph G with 100 nodes. The maximum number of edges to be included in G so that the graph is not connected is
- a. 2451
 - b. 4950
 - c. 4851
 - d. 9900
- 14) Which of the following statements is always correct for any two spanning trees for a graph?
- a. Sum of weight of edges is always same
 - b. Selected vertices have same degree
 - c. Have same number of edges.
 - d. Have same number of edges and sum of weights of edges is also same.
- 15) A connected (all vertices have at least one neighbor), undirected graph of N vertices has (N-1) edges. Number of spanning tree that can be constructed are
- a. 1
 - b. N
 - c. N-1
 - d. $N \times (N-1)$
- 16) A binary tree in which all its levels except the last, have maximum numbers of nodes, and all the nodes in the last level have only one child it will be its left child. Name the tree.
- a. Threaded tree
 - b. Complete binary tree
 - c. M-way search tree
 - d. Full binary tree
- 17) If two trees have same structure and but different node content, then they are called
- _____

- a. Synonyms trees
 - b. Joint trees
 - c. Equivalent trees
 - d. Similar trees
- 18) If two trees have same structure and node content, then they are called _____
- a. Synonyms trees
 - b. Joint trees
 - c. Equivalent trees
 - d. Similar trees
- 19) Which of the following is non-linear data structure?
- a. Stacks
 - b. List
 - c. Strings
 - d. Trees
- 20) To represent hierarchical relationship between elements, which data structure is suitable?
- a. Dequeue
 - b. Priority
 - c. Tree
 - d. Graph
- 21) The number of edges from the root to the node is called _____ of the tree.
- a. Height
 - b. Depth
 - c. Length
 - d. Width
- 22) The number of edges from the node to the deepest leaf is called _____ of the tree.
- a. Height
 - b. Depth
 - c. Length
 - d. Width
- 23) What is a full binary tree?
- a. Each node has exactly zero or two children
 - b. Each node has exactly two children
 - c. All the leaves are at the same level
 - d. Each node has exactly one or two children

- 24) What is a complete binary tree?
- Each node has exactly zero or two children
 - A binary tree, which is completely filled, with the possible exception of the bottom level, which is filled from right to left
 - A binary tree, which is completely filled, with the possible exception of the bottom level, which is filled from left to right
 - A tree in which all nodes have degree 2
- 25) Heap can be used as _____
- Priority queue
 - Stack
 - A decreasing order array
 - Normal Array
- 26) An array consists of n elements. We want to create a heap using the elements. The time complexity of building a heap will be in order of
- $O(n^2 \log n)$
 - $O(n \log n)$
 - $O(n^2)$
 - $O(n \log n \log n)$
- 27) In a max-heap, element with the greatest key is always in the which node?
- Leaf node
 - First node of left sub tree
 - root node
 - First node of right sub tree
- 28) Which of the following algorithms is the best approach for solving Huffman codes?
- exhaustive search
 - greedy algorithm
 - brute force algorithm
 - divide and conquer algorithm
- 29) In Huffman coding, data in a tree always occur?
- roots
 - leaves
 - left sub trees
 - right sub trees

- 30) What is the running time of the Huffman encoding algorithm?
- $O(C)$
 - $O(\log C)$
 - $O(C \log C)$
 - $O(N \log C)$
- 31) The hashing technique which allocates fixed number of buckets is classified as
- Dynamic hashing
 - Static hashing
 - External hashing
 - Internal hashing
- 32) What is the maximum height of an AVL tree with p nodes?
- p
 - $\log(p)$
 - $\log(p)/2$
 - $p/2$
- 33) What is an AVL tree?
- a tree which is balanced and is a height balanced tree
 - a tree which is unbalanced and is a height balanced tree
 - a tree with three children
 - a tree with at most 3 children
- 34) Given an empty AVL tree, how would you construct AVL tree when a set of numbers are given without performing any rotations?
- just build the tree with the given input
 - find the median of the set of elements given, make it as root and construct the tree
 - use trial and error
 - use dynamic programming to build the tree
- 35) Why to prefer red-black trees over AVL trees?
- Because red-black is more rigidly balanced
 - AVL tree store balance factor in every node which costs space
 - AVL tree fails at scale
 - Red black is more efficient

- 36) Which of the following is the most widely used external memory data structure?
- a. AVL tree
 - b. B-tree
 - c. Red-black tree
 - d. Both AVL tree and Red-black tree
- 37) Five node splitting operations occurred when an entry is inserted into a B-tree. Then how many nodes are written?
- a. 14
 - b. 7
 - c. 11
 - d. 5
- 38) 2-3-4 trees are B-trees of order 4. They are an isometric of _____ trees.
- a. AVL
 - b. AA
 - c. 2-3
 - d. Red-Black
- 39) Which of the following is not true about the 2-3 tree?
- a. all leaves are at the same level
 - b. it is perfectly balanced
 - c. postorder traversal yields elements in sorted order
 - d. it is B-tree of order 3
- 40) Which of the following is false?
- a. 2-3 tree requires less storage than the BST
 - b. lookup in 2-3 tree is more efficient than in BST
 - c. 2-3 tree is shallower than BST
 - d. 2-3 tree is a balanced tree
- 41) Which of the following the BST is isometric with the 2-3 tree?
- a. Splay tree
 - b. AA tree
 - c. Heap
 - d. Red – Black tree
- 42) 2-3 tree is a specific form of _____
- a. B – tree
 - b. B+ – tree

- c. AVL tree
 - d. Heap
- 43) A linear list of elements in which deletion can be done from one end (front) and insertion can take place only at the other end (rear) is known as _____
- a. Queue
 - b. Stack
 - c. Tree
 - d. Linked list
- 44) The data structure required for Breadth First Traversal on a graph is?
- a. Stack
 - b. Array
 - c. Queue
 - d. Tree
- 45) A queue follows _____
- a. FIFO (First In First Out) principle
 - b. LIFO (Last In First Out) principle
 - c. Ordered array
 - d. Linear tree
- 46) Circular Queue is also known as _____
- a. Ring Buffer
 - b. Square Buffer
 - c. Rectangle Buffer
 - d. Curve Buffer
- 47) If the elements “A”, “B”, “C” and “D” are placed in a queue and are deleted one at a time, in what order will they be removed?
- a. ABCD
 - b. DCBA
 - c. DCAB
 - d. ABDC
- 48) A data structure in which elements can be inserted or deleted at/from both ends but not in the middle is?
- a. Queue
 - b. Circular queue

- c. Dequeue
 - d. Priority queue
- 49) Which of the following is not the type of queue?
- a. Ordinary queue
 - b. Single ended queue
 - c. Circular queue
 - d. Priority queue
- 50) Queues serve major role in _____
- a. Simulation of recursion
 - b. Simulation of arbitrary linked list
 - c. Simulation of limited resource allocation
 - d. Simulation of heap sort
- 51) Process of inserting an element in stack is called _____
- a. Create
 - b. Push
 - c. Evaluation
 - d. Pop
- 52) Process of removing an element from stack is called _____
- a. Create
 - b. Push
 - c. Evaluation
 - d. Pop
- 53) In a stack, if a user tries to remove an element from an empty stack it is called _____
- a. Underflow
 - b. Empty collection
 - c. Overflow
 - d. Garbage Collection
- 54) Which of the following is not the application of stack?
- a. A parentheses balancing program
 - b. Tracking of local variables at run time
 - c. Compiler Syntax Analyzer
 - d. Data Transfer between two asynchronous process

- 55) Consider the usual algorithm for determining whether a sequence of parentheses is balanced. The maximum number of parentheses that appear on the stack AT ANY ONE TIME when the algorithm analyzes: $((()())())$?
- 1
 - 2
 - 3
 - 4 or more
- 56) Consider the usual algorithm for determining whether a sequence of parentheses is balanced. Suppose that you run the algorithm on a sequence that contains 2 left parentheses and 3 right parentheses (in some order). The maximum number of parentheses that appear on the stack AT ANY ONE TIME during the computation?
- 1
 - 2
 - 3
 - 4 or more
- 57) Here is an infix expression: $4 + 3*(6*3-12)$. Suppose that we are using the usual stack algorithm to convert the expression from infix to postfix notation. The maximum number of symbols that will appear on the stack AT ONE TIME during the conversion of this expression?
- 1
 - 2
 - 3
 - 4
- 58) What is the value of the postfix expression $6\ 3\ 2\ 4\ +\ -\ *?$
- 1
 - 40
 - 74
 - 18
- 59) Entries in a stack are “ordered”. What is the meaning of this statement?
- A collection of stacks is sortable
 - Stack entries may be compared with the ' $<$ ' operation
 - The entries are stored in a linked list
 - There is a Sequential entry that is one by one

60) Pushing an element into stack already having five elements and stack size of 5, then stack becomes _____

- a. Overflow
- b. Crash
- c. Underflow
- d. User flow

61) Which of the following is not a disadvantage to the usage of array?

- a. Fixed size
- b. There are chances of wastage of memory space if elements inserted in an array are lesser than the allocated size
- c. Insertion based on position
- d. Accessing elements at specified positions

62) What is the functionality of the following code?

```
public void function(Node node)
{
    if(size == 0)
        head = node;
    else
    {
        Node temp,cur;
        for(cur = head; (temp = cur.getNext())!=null; cur = temp);
        cur.setNext(node);
    }
    size++;
}
```

- a. Inserting a node at the beginning of the list
- b. Deleting a node at the beginning of the list
- c. Inserting a node at the end of the list
- d. Deleting a node at the end of the list

63) Which of these is not an application of a linked list?

- a. To implement file systems
- b. For separate chaining in hash-tables
- c. To implement non-binary trees
- d. Random Access of elements

- 64) Which of the following is false about a doubly linked list?
- We can navigate in both the directions
 - It requires more space than a singly linked list
 - The insertion and deletion of a node take a bit longer
 - Implementing a doubly linked list is easier than singly linked list
- 65) How do you calculate the pointer difference in a memory efficient double linked list?
- head xor tail
 - pointer to previous node xor pointer to next node
 - pointer to previous node – pointer to next node
 - pointer to next node – pointer to previous node
- 66) What is a memory efficient double linked list?
- Each node has only one pointer to traverse the list back and forth
 - The list has breakpoints for faster traversal
 - An auxiliary singly linked list acts as a helper list to traverse through the doubly linked list
 - A doubly linked list that uses bitwise AND operator for storing addresses
- 67) What differentiates a circular linked list from a normal linked list?
- You cannot have the 'next' pointer point to null in a circular linked list
 - It is faster to traverse the circular linked list
 - You may or may not have the 'next' pointer point to null in a circular linked list
 - Head node is known in circular linked list
- 68) Which of the following is false about a circular linked list?
- Every node has a successor
 - Time complexity of inserting a new node at the head of the list is $O(1)$
 - Time complexity for deleting the last node is $O(n)$
 - We can traverse the whole circular linked list by starting from any point
- 69) Consider a small circular linked list. How to detect the presence of cycles in this list effectively?
- Keep one node as head and traverse another temp node till the end to check if its 'next' points to head
 - Have fast and slow pointers with the fast pointer advancing two nodes at a time and slow pointer advancing by one node at a time

- c. Cannot determine, you have to pre-define if the list contains cycles
- d. Circular linked list itself represents a cycle. so no new cycles cannot be generated

70) What is the functionality of the following code? Choose the most appropriate answer.

```
public int function()
{
    if(head == null)
        return Integer.MIN_VALUE;
    int var;
    Node temp = head;
    Node cur;
    while(temp.getNext() != head)
    {
        cur = temp;
        temp = temp.getNext();
    }
    if(temp == head)
    {
        var = head.getItem();
        head = null;
        return var;
    }
    var = temp.getItem();
    cur.setNext(head);
    return var;
}
```

- a. Return data from the end of the list
- b. Returns the data and deletes the node at the end of the list
- c. Returns the data from the beginning of the list
- d. Returns the data and deletes the node from the beginning of the list

71) The given array is arr = {1, 2, 4, 3}. Bubble sort is used to sort the array elements. How many iterations will be done to sort the array?

- a. 4
- b. 2

- c. 1
- d. 0

72) Which of the following is not an advantage of optimised bubble sort over other sorting techniques in case of sorted elements?

- a. It is faster
- b. Consumes less memory
- c. Detects whether the input is already sorted
- d. Consumes less time

73) How many passes does an insertion sort algorithm consist of?

- a. N
- b. N-1
- c. N+1
- d. N^2

74) What will be the number of passes to sort the elements using insertion sort?

14, 12, 16, 6, 3, 10

- a. 6
- b. 5
- c. 7
- d. 1

75) For the following question, how will the array elements look like after second pass?

34, 8, 64, 51, 32, 21

- a. 8, 21, 32, 34, 51, 64
- b. 8, 32, 34, 51, 64, 21
- c. 8, 34, 51, 64, 32, 21
- d. 8, 34, 64, 51, 32, 21

76) Which of the following sorting algorithms is the fastest for sorting small arrays?

- a. Quick sort
- b. Insertion sort
- c. Shell sort
- d. Heap sort

77) On which algorithm is heap sort based on?

- a. Fibonacci heap
- b. Binary tree

- c. Priority queue
 - d. FIFO
- 78) How many arrays are required to perform deletion operation in a heap?
- a. 1
 - b. 2
 - c. 3
 - d. 4
- 79) Which of the following sorting algorithms is the fastest?
- a. Merge sort
 - b. Quick sort
 - c. Insertion sort
 - d. Shell sort
- 80) How do you initialize an array in C?
- a. `int arr[3] = (1,2,3);`
 - b. `int arr(3) = {1,2,3};`
 - c. `int arr[3] = {1,2,3};`
 - d. `int arr(3) = (1,2,3);`
- 81) How is time complexity measured?.
- a. By counting the number of algorithms in an algorithm.
 - b. By counting the number of primitive operations performed by the algorithm on given input size.
 - c. By counting the size of data input to the algorithm.
 - d. By counting steps in given algorithm.
- 82) The complexity of merge sort algorithm is
- a. $O(n)$
 - b. $O(\log n)$
 - c. $O(n^2)$
 - d. $O(n \log n)$
- 83) The number of swappings needed to sort the numbers 8,22,7,9,31,5,13 in ascending order, using bubble sort is
- a. 11
 - b. 12
 - c. 13
 - d. 10

- 84) ----- analysis is use to find out which algorithm is better for the same problem statement.
- Big O notation
 - Theta notation
 - Asymptotic
 - Omega notation
- 85) The data structure which stored data of similar type is known as -----data structure.
- Dynamic
 - Homogenous
 - Non homogenous
 - Linear
- 86) Dividing the list into two or more lists as per requirement of the program
- Sorting
 - Splitting
 - Creation
 - Merging
- 87) -----array is a special type of 2 dimensional array consist of m rows and n columns.
- Multidimensional
 - Merging
 - Sparse
 - Vector
- 88) A -----is a linear data structure which stores the elements in a node in a linear fashion but not necessarily contiguous.
- Array
 - LinkedList
 - Pointer
 - Reversing
- 89) The function -----reserves a block of memory of specified size and returns a pointer of type void which can casted into pointer of any form.
- Realloc()
 - Free()
 - Calloc()

d. Malloc()

90) If you are looking for the number which is at the end of the list then you need to search entire list which is time consuming. Disadvantage of

- a. Binary Search
- b. Bubble Sort
- c. Linear Search
- d. Quick Sort

91) Big Oh Notation (O)

- a. $f(n) \geq c \cdot g(n)$ for all $n \geq n_0$ and $c > 0$
- b. $f(n) \leq c \cdot g(n)$ for all $n \geq n_0$
- c. $c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n)$
- d. $f(n) = \theta(g(n))$

92) A _____ is a group of instructions that performs a specific given task when executed by a computer.

- a. Algorithm
- b. Program
- c. Data Structure
- d. Data

93) What will be the output of this program?

```
int main()
{
int a=10, b=20;
printf("a=%d b=%d",a,b);
a=a+b;
b=a-b;
a=a-b;
printf("a=%d b=%d",a,b);
return 0;
}
```

- a. a = 20, b = 20
- b. a = 10, b = 20
- c. a = 20, b = 10
- d. a = 10, b = 10

94) Predict the output of below code:

```
#include "stdio.h"
#include "stdlib.h"

int main(int argc, char *argv[]) {
    char temp[20];
    gcvt(23.45,2, temp);
    printf("%s", temp);
    return 0;
}
```

- a. 0.4
- b. 24.4
- c. 25.4
- d. 23

95) What is required in each C program?

- a. The program must have at least one function.
- b. The program does not require any function.
- c. Input data
- d. Output data

96) What will this program print?

```
main()
{
    int i = 2;
    {
        int i = 4, j = 5;
        printf("%d %d", i, j);
    }
    printf("%d %d", i, j);
}
```

- a. 4525
- b. 2525
- c. 4545

d. 4455

97) What is the result after execution of the following code if a is 10, b is 5, and c is 10?

```
If ((a > b) && (a <= c))
```

```
    a = a + 1;
```

```
else
```

```
    c = c+1;
```

a. a = 10, c = 10

b. a = 11, c = 10

c. a = 10, c = 11

d. a = 11, c = 11

98) How many times will the following loop execute?

```
for(j = 1; j <= 10; j = j-1)
```

a. Forever

b. Never

c. 0

d. 1

99) What will be the output of this program?

```
int main()
```

```
{
```

```
int a=10, b=20;
```

```
printf("a=%d b=%d",a,b);
```

```
a=a+b;
```

```
b=a-b;
```

```
a=a-b;
```

```
printf("a=%d b=%d",a,b);
```

```
return 0;
```

```
}
```

a. a = 20, b = 20

b. a = 10, b = 20

c. a = 20, b = 10

d. a = 10, b = 10

100) Which of these describe an array?

- a. A data structure that shows a hierarchical behavior
- b. Container of object of similar types
- c. Arrays are immutable once initialized
- d. Array is not a data structure