Subject: **Data Structures CS301**
Test: **Midterm past Paper questions**
Total Questions: **91**

Question No.	Question Details	Total Marks
1	☑ ddd	
dfdfdf		
dfd		ddd ddd dfdf df dfd f d f d f
1	2 ☑	1
2		

3			Deletin g a leaf nod e in bin ary sea rch tree inv olv es sett ing ____ ____ poi nter /s of that nod e's par ent as null .
1	3	☐	Tru e

False

	A node cannot be deleted, when the node to be deleted has both left and right subtrees.
	True
	False

1	4	☑	True
False			
	An array is a group of consecutive related memory locations.		
	True		
	False		
1	5	☐	3 6 5 + * 7 5 8 + - *
3 6 5 7 5 8 + * + - *			

3 5 6 + * 7 8 5 + - *

Consider the following infix expression: 3 + 5 * 6 – 7 * (8 + 5) Which of the following is a correct equivalent expression(s) for the above?

1	6	☐	two children per node whereas a binary tree can have none, one, or two children per node

in binary search tree nodes are inserted based on the values they contain

in binary tree nodes are inserted based on the values they contain

The difference between a binary tree and a binary search tree is that ,a binary search tree has

two children per node whereas a binary tree can have none, one, or two children per node

in binary search tree nodes are inserted based on the values they contain

in binary tree node are inserted based on the value they contain

none of these

1	7	☐	Front

Rear

From Both Rare and Front

We can add elements in QUEUE From ____ ____ ____

Front

Rear

From Both Rare and Front

None of these

1	8	☐	short

Int

float	Which of the following abstract data types are NOT used by Integer Abstract Data type group?
	short
	Int
	float long

1	9	☑	Array size is fixed once it is created

Link List size is fixed once it is created

Binary Search Tree size isfixed once it is created	Which one of the following statements is correct?
	Array size is fixed once it is created
	Link List size is fixed once it is created

			Binary Search Tree size is fixed once it is created
			AVL Tree size is fixed once it is created
1	10	☐	True

False

	Linked lists are collections of data items "lined up in a row" , insertions and deletions can be made only at the front and the back of a linked list.
	True
	False

1	11	☑	int &x ;
int *x ;			

int x ;

In a program a reference variable, say x, can be declared as

int &x ;

int *x ;

			int x; None of the given options
1	12	☑	In linked list the elements are necessarily to be contiguous
In linked list the elements may locate at far positions in the memory			

In linked list each element also has the address of the element next to it	Which one of the following statement is NOT correct.
	In linked list the elements are necessarily to be contiguous

In linked list the elements may locate at far positions in the memory

In linked list each element also has the address of the element next

In an array the elements are contiguous

			to it
1	13	☐	Array
List			

Both of these	Is a data structure that can grow easily dynamically at runtime without having to copy existing elements?
	Array
	List

			Both of these / None of these
1	14	☐	Last in First out
First in Last out			

First in First out

Queue follows

Last in First out

First in Last out

First in First out

None of these

1	15	☑	Stack

Queue

Binary Search Tree	Parameters in function call are passed using,
	Stack
	Queue
	Binary Search Tree / AVL Tree

1	16	☐	Lists can be implemented by using arrays or linked lists
A list is a sequence of one or more data items			

Stack is a special kind of list in which all insertions and deletions take place at one end	Which statement of the following statements is incorrect?
	Lists can be implemented by using arrays or linked lists

A list is a sequence of one or more data items

	Stack is a special kind of list in which all insertions and deletions stak	Stacks are easier to implement than lists

			replaceatoneend
1	17	☐	delete
peek			

pop

The operation for removing an entry from a stack is traditionally called:

- delete
- peek
- pop
- remove

1	18	☐	That pointer points to First byte in main function
That pointer contains a NULL value			

That pointer points to any memory address

Suppose a pointer has been declared in main but has not assigned any variable address then

	That pointer points to First byte in main function
	That pointer contains a NULL value

			That pointer points to any memory address	None of these
1	19	☑	Arrays	
LinkLists				

Binary Search Trees

A Compound Data Structure is the data structure which can have multiple data items of same type or of different types. Which

of the following can be considered compound data structure?

Arrays

LinkLists

			Binary Search Trees All of the given options
1	20	☐	Linear data structure
Non-linear data structure			

Graphical data structure	The tree data structure is a
	Linear data structure
	Non-linear data structure

			Graphical data structure	Data structure like queue
1	21	☐	ab+c*d-	

abc*+d-

abc+*d-

Which one of the following is a valid postfix expression?

ab+c*d-

abc*+d-

abc+*d-

(abc*)+d-

1 22

Prefix

Postfix

Infix

When an operator is used in between two operands this is which type of notation

Prefix

Postfix

		Infix	None of the Above
1	23	☐	ABC+*D-
ABC*+D-			

ABCD+-*			What will be the valid postfix notation of A+ B* C- D
			AB C+* D-
			AB C*+ D-
			A B C D +- * | A B + D * C
1	24	☐	Linear
Non Linear			

Circular			Tree data structure is a
			Linear
			Non Linear
			Circular / None of Above
1	25	☑	True
False			
	Non recursive calls are faster than the Recursive calls.		
	True		
	False		
1	26	☐	Stacks
Queues			

Both Stacks and Queues

Following are the linear data structures:-

Stacks

Queues

Both Stacks and Queues

None of the above

1 27

Plus

Minus

Multiply			Highest Operators Precedence is of the following operator:-
			Plus
			Minus
			Multiply / Exponentiation
1	28	☐	1
2			

3			Each node in a BST has Pointers:- 1 2 3 4
1	29	☐	Size can be increase but can not be decreased.
Size can be decreased but can not be increased.			

Size can neither be increased nor be decreased.	After creation of an array:-
	Size can be increase but cannot be decreased.
	Size can be decreased but cannot be increased.

			Size can neither be increased nor be decreased.	Size can be increased and can also be decreased
1	30	☐	Linear	
Non Linear				

Circular			BST is a Structure:-
			Linear
			Non Linear
			Circular / Non of Above
1	31	☐	1.66 Log2n
1.44 Log2n			

Log2 (n+1)-1	To check the depth of an AVL tree following time will be taken:-
	1.66 Log2n
	1.44 Log2n

			Log2(n+1)-1	1.66Log2n(n+1)
1	32	☐	Single	
Double				

Triple	In an AVL tree to delete a parent with two childs in a straight line following rotations will be required
	Single
	Double

		Triple	None.of the given options
1	33	☐	Linked List
Stack			

Queue			Which of the following is a non linear data structure?
			Linked List
			Stack
			Queue / Tree
1	34	☐	56/ + 2
5 6 2 / +			

/62 + 5

Consider the following infix expression. 5 + 6/2 If one converts the above expression into postfix, what would be the resultant expressio

			n?
			56/ + 2
			5 6 2 / +
			/6 2 + 5 | 5 6 / 2 +
1	35	☐	1
2			
3			There is/are case/s for rotation in an AVL tree,
			1
			2
			3 | 4

1	36	☐	Log 2(n +1) tim e (wh ere n is no. of nod es
Log2(n+1) -1			

1.44 Log2n	Searching an element in an AVL tree take maximum in AVL tree,
	Log2(n+1) time (where n is no. of nodes
	Log2(n+1)-1

			1.44Log2n	1.66Log2n
1	37	☐	We can increase the size of arrays after their creation.	
We can decrease the size of arrays after their creation.				

We can increase but can't decrease the size of arrays after their creation.	Which of the following is "TRUE" about arrays, We can increase the size of arrays after their creation.

We can decrease the size of arrays after their creation.

We can increase but can't decrease the size of array after their cre

We can neither increase nor decrease the array size

		ation.	after creation.
1	38	☐	Trees are recursively defined multi-dimensional data structures tree

The order of a tree indicates a maximum number of children allowed at each node of the

A search tree is a special type of tree where all values (i.e. keys) are ordered

Four statements about trees are below. Three of them are correct. Which one is INCORRECT?

Trees are recursively defined multi-dimensional data structures tree

The order of a tree indicates a maximum number of children allowed at each node of the

A search tree is a special type of tree where all values (i.e. keys) are

If Tree1's size is greater than Tree2's size, then

ordered

the height of Tree1 must be greater than Tree2

			'sheight.
1	39	☑	Multiplication operator
Minus operator			

Plus operator	Which one of the following operators has higher priority than all of others?
	Multiplication operator
	Minus operator

			Plus operator	Exponentiation operator
1	40	☑	FIFO, LIFO	

LIFO,FIFO

both given options	A queue is a------ data structure, whereas a stack is a ----- data structure.
	FIFO, LIFO
	LIFO,FIFO

			both given options / None of these
1	41	☐	if (alpha < beta)

if (alpha.LessThan(beta))

```
if (LessThan(alpha, beta))
```

Suppose that the class declaration of SomeClass includes the following function prototype. bool LessThan(SomeClass anotherObject);

	Which of the following tests in the client code correctly compares two class objects alpha and beta?
	if (alpha < beta)

```
if (alpha.LessThan(beta))
if (LessThan(alpha, beta))
if (LessThan(alpha).beta)
```

1	42	☑	Array size can be changed after its creation.

Link List size can be changed after its creation

Binary Search Tree size can be changed after its creation	Which one of the following statements is NOT correct?
	Array size can be changed after its creation.

Link List size can be changed after its creation

	Binary Search Tree size can be changed after its creation	AVL Tree size can be changed after its creation

1	43	☐	None of the given options

Call by passing the value of the argument

Call by passing reference of the argument	Which one of the following calling methods does not change the original value of the argument in the calling function?

None of the given options

Call by passing the value of the argument

			Call by passing reference of the argument	Call by passing the address of the argument
1	44	☑	The first element	

The middle element

The last element	In an array list the current element is
	The first element
	The middle element

			The last element	The element where the cursor points to
1	45	☐	True	

False			
	In an array we can store data elements of different types.		
	True		
	False		
1	46	☑	True
False			
	An array is a group of consecutive related memory locations.		
	True		
	False		
1	47	☐	(iii) and (ii) only
(i), (ii) and (iv) only			

(ii) and (iv) only

The following are statements related to queues. (i) The last item to be added to a queue is the first item to be removed (ii) A queue is a stru

cture in which both ends are not used (iii) The last element has n't to wait until all elements preceding it on the queue are removed (iv) A

queue is said to be a last-in-first-out list or LIFO data structure. Which of the above is/are related to normal queues?

(iii) and (ii) only

			(i), (ii) and (iv) only
			(ii) and (iv) only
			None of the given options
1	48	☑	Stack
Queue			

Both of these

Which of the following can be used to reverse a string value,

Stack

Queue

Both of these

None of these

1	49	☐	Zero

2n (where n is the number of nodes in linked list)

Any Number	_____ is the maximum number of nodes that you can have on a stack-linked list?
	Zero

			2n (where n is the number of nodes in linked list)
			Any Number / None of these
1	50	☑	7 8 9 10 11 12

9 8 11 10 7 12

9 10 8 11 12 7

Consider the following sequence of push operations in a stack: stack.push('7'); stack.push('8'); stack.push('9'); stack.push('10'); stack.pus

			h('11'); stack.push('12');
			7 8 9 10 11 12
			9 8 11 10 7 12
			9 1 0 8 1 1 1 2 7 9 1 0 8 1 1 2 7 1
1	51	☐	delete
peek			

pop

The operation for removing an entry from a stack is traditionally called:

delete

peek

pop

remove

1 52

add

append

insert	The operation for adding an entry to a stack is traditionally called : add append insert push

1	53	☐	Assign a value to primitive type using a literal
Declare primitive types to be constant using the Const keyword			

Create a new instance of primitive type with New keyword	In C what is the operation that you can not do with primitive types?
	Assign a value to primitive type using a literal

Declare primitive types to be constant using the Const keyword

			Create a new instance of primitive type with New keyword	None of these
1	54		if (alpha < beta)	

```
if (alpha.LessThan(beta))
```

```
if (LessThan(alpha, beta))
```

Suppose that the class declaration of SomeClass includes the following function prototype. bool LessThan(SomeClass anotherObject);

	Which of the following tests in the client code correctly compares two class objects alpha and beta?
	if (alpha < beta)

			if (alpha.LessThan(beta))
			if (LessThan(alpha, beta))
			if (LessThan(alpha).beta)
1	55	☑	(currentNode == null)
(currentNode->nextNode == null)			

(nextNode.data == null)

Suppose current Node refers to a node in a linked list (using the Node class with member variables called data and nextNode). What boo

	lean expression will be true when cursor refers to the tail node of the list ?
	(currentNode == null)
	(currentNode->nextNode == null)

			(nextNode.data==null) (currentNode.data==0.0)
1	56	☐	Any one node fulfills the AVL condition
At least half of the nodes fulfill the AVL condition			

All the nodes fulfill the AVL condition

A tree is an AVL tree if

Any one node fulfills the AVL condition

At least half of the nodes fulfill the AVL condition

			All the nodes fulfill the AVL condition	None of the given options
1	57	☑	In preorder traversal only	

In inorder traversal only

In postorder traversal only	In which of the traversal method, the recursive calls can be used to traverse a binary tree?
	In preorder traversal only

			In inorder traversal only
			In postorder traversal only
			All of the given options
1	58	☐	True

False

	Doubly Linked List always has one NULL pointer.
	True
	False

1	59	☑	True

False

	A subscript of an array may be an integer or an integer expression.
	True
	False

1	60	☐	Not an
Unary			
Binary			“+” is a _____ operator.
			Not an
			Unary
			Binary / Ternary
1	61	☐	Not an
Unary			

Binary			“+” is a _____ operator.
			Not an
			Unary
			Binary / Ternary
1	62	☐	One
Two			

Three			A binary search tree should have minimum of one ____ ____ __ node/s at each level,
			One
			Two
			Three / Four
1	63	☐	Linear way only
Non Linear way only			

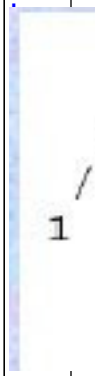
Both linear and non linear ways	We access elements in AVL Tree in,
	Linear way only
	Non Linear way only

			Both linear and non linear ways
			None of the given options
1	64	☐	Root Nodes
Leaf Nodes			

Both of these			The nodes with no successor are called ____	
			Root Nodes	
			Leaf Nodes	
			Both of these	None of these
1	65	☐	8	
7				

5

Consider the following tree.



How many of the nodes have at least one sibling?

8

7

5 6

1	66	☐	Neither changes

Only front pointer changes.

Only rear pointer changes.

I have implemented the queue with a linked list, keeping track of a front pointer and a rear pointer. Which of these pointers will cha

nge during an insertion into an EMPTY queue?

Neither changes

Only front pointer changes.

			Only rear pointer changes.	Both change.
1	67	☐	1 pointer	
2 pointer				

3 pointer			Each node in doubly link list has ,
			1 pointer
			2 pointer
			3 pointer / 4 pointer
1	68	☐	Stack
Queue			

Link list			Which one is a self-referential data type?
			Stack
			Queue
			Link list
			All of these
1	69	☐	False
True			
	A queue where the de-queue operation depends not on FIFO, is called a priority queue		
	False		
	True		
1	70	☐	currentNode++;
currentNode = nextNode;			

currentNode += nextNode;

Suppose current Node refers to a node in a linked list (using the Node class with member variables called data and nextNode). What stat

	ement changes current Node so that it refers to the next node?
	current Node ++;
	current Node = nextNode;

```
currentNode+=nextNode;
currentNode=currentNode->nextNode;
```

1	71	☐	Any one node fulfills the AVL condition
At least half of the nodes fulfill the AVL condition			

All the nodes fulfill the AVL condition (	A tree is an AVL tree if
	Any one node fulfills the AVL condition
	At least half of the nodes fulfill the AVL condition

			All the nodes fulfill the AVL condition (None of the given options
1	72	☐	None of the given options

Call by passing the value of the argument

Call by passing reference of the argument	Which one of the following calling methods does not change the original value of the argument in the calling function?

None of the given options

Call by passing the value of the argument

			Call by passing reference of the argument	Call by passing the address of the argument
1	73	☐	One	
Two				

Three	Each operator in a postfix expression refers to the previous _____ operand(s).
	One
	Two
	Three / Four

1	74	☑	In linked list the elements are necessarily to be contiguous

In linked list the elements may locate at far positions in the memory

In linked list each element also has the next to it	Which one of the following statement is NOT correct.
	In linked list the elements are necessarily to be contiguous

In linked list the elements may locate at far positions in the memory

			In linked list each element also has the next to it	In an array the elements are contiguous
1	75	☑	Non Linear data structure	

Linear data structure

Hybrid data structure (Mixture of Linear and Non Linear)	AVL Tree is, Non Linear data structure Linear data structure

			Hybrid data structure (Mixture of Linear and Non Linear)	None of the given options.
1	76			Linear way only

Non Linear way only

Both linear and non linear ways	We access elements in AVL Tree in,
	Linear way only
	Non Linear way only

			Both linear and non linear ways	None of the given options.
1	77	☐	J and I	
H and E				

D and E

Consider the following binary search tree (BST):

If node A in the BST is deleted, which two nodes are the candidates to tak

			e its place?
			J and I
			H and E
			D and E / L and M
1	78	☐	Strictly Binary Tree
Binary Search tree			

AVL tree

___
___
___
is a bin ary tree wh ere eve ry nod e has a val ue, eve ry nod e's left sub tree con tain s onl y val ues les s tha n or equ al to the nod e's val

	ue, and every node's right subtree contains only values that are greater then or equal?
	Strictly Binary Tree
	Binary Search tree

			AVL tree	All of these
1	79	☐	Prefix expression	
Postfix expression				

Infix expression

The expression AB+C* is called?

Prefix expression

Postfix expression

Infix expression

None of these

1	80	☐	data[1]

data[2]

data[11]

Suppose we have a circular array implementation of the queue class, with ten items in the queue stored at data[2] through data[11]. The CAPA

CITY is 42, i.e., the array has been declared to be of size 42. Where does the push member function place the new entry in the array?

			data[1] data[2] data[11] data[12]
1	81	☑	At the head
At the tail			

After all other entries that are smaller than the new entry.	In the linked list implementation of the stack class, where does the push member function places the new entry on the linked list?

	At the head
	At the tail

After all other entries that are smaller than the new entry.

After all other entries that are greater than the new

			w e n t r y .
1	82	☑	Log 2 (n+ 1) - 1
Log2 (n+1)			

Log2 (n) – 1	Suppose n is the number of nodes in a complete Binary Tree then maximum steps required for a search operation are,
	Log 2 (n+ 1) - 1

			Log 2 (n+ 1)
			Log 2 (n) – 1 / Log 2 (n)
1	83	☐	1
2			

n (where n is the argument)	What is the maximum depth of recursive calls a function may make?
	1
	2

	n (where n is the argument)	There is no fixed maximum

1	84	☐	Only x can alter the private member variables of the object that activates the function.

Only y can alter the private member variables of the object that activates the function.

Only z can alter the private member variables of the object that activates the function.

Here is the start of a C++ class declaration:
```
class foo {
public:
void x(foo f);
void y(const foo f);
void z(foo f) const;
...
```
Which of the

three member functions can alter the PRIVATE member variables of the foo object that activates the function?

Only x can alter the private member variables of the object that activates the function.

Only y can alter the private member variables of the object that activates the function.

Only can alter the private member variables of the object that activates

Two of the function prototypes can have member

the function.

variables of the objective function

			n.
1	85	☐	Whenever the parameter has huge size.

Whenever the parameter has huge size, the function changes the parameter within its body, and you do NOT want these changes to alter the actual argument.

Whenever the parameter has huge size, the function changes the parameter within its body, and you DO want these changes to alter the actual argument.	When should you use a const reference parameter?
	Whenever the parameter has huge size.

Whenever the parameter has huge size, the function changes the parameter within its body, and you do NOT want these changes to alter

the actual argument.

Whenever the parameter has shuge size, the function changes the p

Whenever the parameter has shuge size, and the func

	arameter within its body, and you DO want these changes to alter th	tion does not change the parameter within its body

			e a ct u al ar g u m e nt . .
1	86	☐	Lin ear dat a stru ctur e
Non-linear data structure			

Graphical data structure	The tree data structure is a
	Linear data structure
	Non-linear data structure

			Graphical data structure Data structure like queue
1	87	☑	True
False			
	In the call by value methodology, a copy of the object is passed to the called function.		
	True		
	False		
1	88	☐	a = X (b) ;
a = X (&b) ;			

a = X (*b) ;

Consider the function X as under int X (int & Value) { return Value; } Now a and b are integers in a calling function. Which one of the foll

owing is a valid call to the above function X.

a = X (b) ;

a = X (&b) ;

		a = X (* b) ;	None of the given options
1	89	☐	Use better data structures
Increase the hard disk space			

Use the better algorithm	The data of the problem is of 2GB and the hard disk is of 1GB capacity, to solve this problem we should
	Use better data structures

	Increase the hard disk space

			Use the better algorithm	Use as much data as we can store on the hard disk
1	90	☑	True	
False				

	A queue where the de-queue operation depends not on FIFO, is called a priority queue		
	True		
	False		
1	91	☐	No option found!
No option found!			
	No Question Found		
	No option found!		
	No option found!		

Signature:

_____