

CS301 Updated 180 MCqs

Quiz 1 Lec 1-8 Spring 2022

**#Midterm Preparation Most
repeated MCqs**

REMEMBER OWN PRAYER

Orange Monkey Team

Important NOTE:-

**Like share and comment too ..ager koi
mistake hogaye ho to sorry and comment
main lazmi mention krdijiye ga takey main
correction krlon file main And like
zaroor krein is hamein or bhe apsab
keyliye files makes krny ke himat milti hey
...JAZAKALLAH**

Orange Monkey Team



1. In array list the worst case of removing an element is.

To remove an element from the end of the list.....confirm

In array list the (Raise For Success) worst case of removing an element is:

- To remove an element from the end of the list Correct

2. End() method of list performs it's tasks in

One step.....confirm

position just in one step. The length() method returns the value of the size of the list. The other methods *back()*, *start()* and *end()* also perform their tasks only in one step.

3. In singly linked list “next” field of node contains

Adderss of the next node....confirm

- u. none of the given
In singly linked list "next" field of node contains:
a. Address of next node

4. If we do not have to move too much in the list then use of which one type of list is not necessary/essential

Circular linked list.....confirm

advantages depend on its use. If we do not have to move too much in the list and have no problem checking the NULL, there is little need a circularly-linked list. But this facility is available to us.

5. Null is an invalid address and _____.

Inaccessible....confirm

memory address NULL in the last node. NULL is an invalid address and is inaccessible.

6. Suppose a stack class has been defined using template. Now, we want to declare a stack object of an int type. What will be the correct syntax.

Stack<int> stack....confirm

In the above code, consider the line

Stack <int> intstack ;

This line means that while creating an object *intstack* of Stack, the generic data type *T*

7. Which off the following is not a form of expression?

Pastfix....confirm

Infix, Postfix and Prefix notations are three different but equivalent ways of writing expressions. It is easiest to demonstrate the differences by looking at examples of

8. The back() method decreases the value of variable current by

One....confirm

The `back()` method decreases the value of variable `current` by 1.

9. 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.....confirm

10. In the linked list implementation of the stack class, where does the push member function places the new entry on the linked list?

A. At the head ✓

10. Whenever we call a function, the compiler makes a _____ that it uses to fulfill this function call.

Stack.....confirm

domain compiler. Whenever we call a function, the compiler makes a stack that it uses to fulfill this function call. The compiler puts the entries on the stack in the way

11. Which of the following is not a Data structure?

Memory cell confirm

12. Compiler uses which one of the following in functions call.

Stack....confirm

Question 1: Compiler uses which one of the following in Function calls _____.



Stack



Queue



Binary



AVL Tree

Search Tree

13. Which one option is associated with the data structures?

Time to perform each basic operation.....confirm

There are three basic things associated with data structures. A data structure requires:

- space for each data item it stores
- time to perform each basic operation
- programming effort

14. Which of the following line of code is incorrect?

`Float *I = float new`

15. Which kind of data structure is used to evaluate postfix expression?

Stack.....confirm

We are going to evaluate the postfix expression with the help of stack.

16. We cannot remove items randomly from _____.

Stack and queue....confirm

125. We can not remove items randomly from _

- a) Stack
- b) Queue
- c) Both of these

17. Length() method of list class is used to __.

Return the length of the list.....confirm

s `length()` that simply returns the size of the list.

18. A template is a function or class that is written with a _____ data type

Generic.....confirm

in C++ language, a **template is a** function or class that is written with a generic data type. When a programmer uses this function or class, the generic data type is replaced

19. Each operator in a postfix expression refers to the previous _____ operand(s).
One....confirm

Each **operator in a postfix expression** refers to the previous _____ operand(s).

1. **One**

20. The principal benefit of a linked list over a conventional array is that the order of the linked items may be _____
Same...confirm

The **principal** benefit of a linked list over a conventional array is that the order of the linked items may be _____ from the order that the data items are stored in memory.

Same

21. IsEmpty() method of stack class will return true when _____
Stack is empty....confirm

scenarios, what happens if we can *pop()* while there is no element? One possible way-out is that we have *isEmpty()* function that returns true **if stack is empty** and false otherwise. This is a boolean function that returns false if there is no element in the

22. End() method of list performs it's task in _____
One step.....confirm

end() also perform their tasks only in one step.

23. In queue data structure elements are removed from _____.

Front....confirm

Queues are data structures that follow the First In First Out (FIFO) i.e. the first element that is added to the queue is the first one to be removed. **Elements are always added to the back and removed from the front.** Think of it as a line of people waiting for a bus.

24. A software solution is said to be efficient if it solves the problem

Within its resource constraints....confirm

A solution is said to be efficient if it solves the problem **within the required resource constraints**. Examples of resource constraints include the total space available to store th

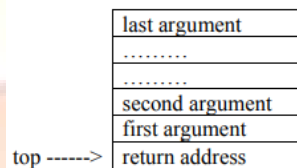
25. Generalized code written for a class is called.

Template....confirm

this class so far. First we define the class `BinaryNode` and its nodes, a binary tree is created. This **class is a template class**.

26. Whenever we call a function the compiler makes a stack the top element of the stack is _____ of the function

Last argument....confirm



27. New int[11] will allocate memory for _____ integers.

11...confirm

"new int[11]" **will allocate** memory for _____ integers.

1. 10
2. **11**

28. At the time while evaluating an expression, if we want to give higher precedence to lower precedence operator, we used.

Parenthesis.....confirm

parenthesis in the infix especially at the times when we want to give a higher precedence to an operator of **lower precedence**. For example, if there is a + operator

29. "+" is a _____ operator.

Binary.....confirm

Binary operator who hoty hein jo 2 operand key sath use hoty hon ..like $A + B$..idher A or B operand hein or + operator hey.. or Uniary symbol just one operand key sath use hoty hein like $(- -a)$ idher - - ye operator hey or a operand hein

30. The stack implementing an array causes worst case behavior when the insertion and deletion of an element done from

Beginning of an array.....confirm

that the stack is internally implemented with the help of array. The worst case for insertion and deletion from an array may happen when we insert and delete from the beginning of the array. We have to shift elements to the right for insertion and left for

31. Each node in singly linked list contains _____.

One pointer.....confirm

Each node in singly link list has,

1. 1 pointer

32. $2*(7+8)-6$ is an example of ____ expression

Infix.....confirm

how can we convert the infix expression $A + (B * C)$

33. If we use doubly linked list to implement list to implement list then there is an issue of;

Pre pointer of first node and next pointer of last node are

NULL.....confirm

pointers, we can move forward and backward very easily. Now we face another problem that the prev pointer of first node and the next pointer of the last node are NULL. Therefore, we have to be careful in case of NULL pointers. To remove the

34. Allocating and de-allocating memory for linked list nodes does take _ time than pre-allocated array.

More.....confirm

Allocating and de-allocating memory for list nodes **does take** more time than pre-allocated array. Memory allocation and de-allocation has cost in terms of time.

35. Circular linked list always has _ NULL pointer/s in a node.

0.....confirm

A circular list is a list that does not contain any pointer pointing to NULL.

36. The type of expression in which operator succeeds its operands is _____ expression.

Postfix.....confirm

postfix expression

Explanation: The expression in which operator succeeds its operands is called **postfix expression**.

37. Before using the push method of stack the user must call the _____ method.

isFull(); confirm

Before using the (Raise For Success) push method of stack, the user must call the _____ method.

- isEmpty();
- **isFull(); Correct**

38. What will be postfix expression of following infix expression?

$4 * 1 \uparrow 7 + 2$

$41 \uparrow 7 * 2 +$ confirm

fix: $4*1\uparrow7+2$

Infix to Postfix

stfix: $41\uparrow7*2+$

39. If the input symbol is not ')' and the stack is not empty then which of the following option will be pushed on the stack.

Operator.....confirm

If the input symbol is not ')' and the stack is not empty, we will push the operator on the stack. Otherwise, it is advisable to pop the stack and discard the '('. The following

40. There is no such node whose next field is NULL, which one of the given option supports the statement .

Circular link list.....confirm

41. Trying to remove an element from an empty stuck is called _____.

Underflow of stack....confirm

Trying to remove an element from an empty stack is called _____.

1. Garbage collector
2. Overflow of stack
3. Empty collection
4. Underflow of stack

42. In doubly linked list a node consists of three parts:

2 pointers and 1 object...confirm

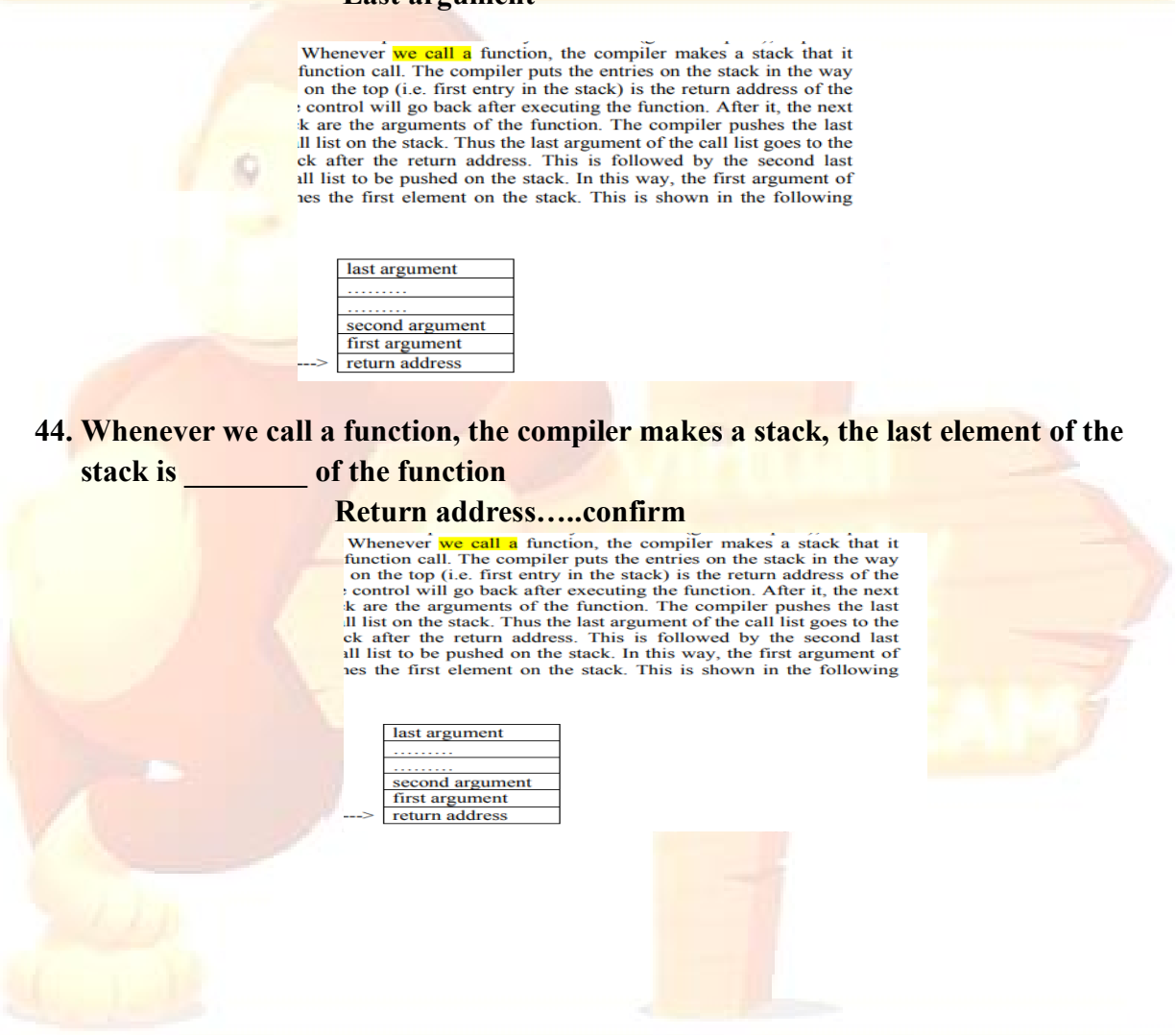
In doubly linked **list a node** consists of three parts:

2 pointers and 1 object

43. Whenever we call a function, the compiler makes a stack, the top element of the stack is _____ of the function

Last argument

Whenever **we call a** function, the compiler makes a stack that it function call. The compiler puts the entries on the stack in the way on the top (i.e. first entry in the stack) is the return address of the ; control will go back after executing the function. After it, the next k are the arguments of the function. The compiler pushes the last ll list on the stack. Thus the last argument of the call list goes to the ck after the return address. This is followed by the second last all list to be pushed on the stack. In this way, the first argument of es the first element on the stack. This is shown in the following



last argument
.....
.....
second argument
first argument
return address

44. Whenever we call a function, the compiler makes a stack, the last element of the stack is _____ of the function

Return address.....confirm

Whenever **we call a** function, the compiler makes a stack that it function call. The compiler puts the entries on the stack in the way on the top (i.e. first entry in the stack) is the return address of the ; control will go back after executing the function. After it, the next k are the arguments of the function. The compiler pushes the last ll list on the stack. Thus the last argument of the call list goes to the ck after the return address. This is followed by the second last all list to be pushed on the stack. In this way, the first argument of es the first element on the stack. This is shown in the following

last argument
.....
.....
second argument
first argument
return address

45. Consider the linked list implementation of a stack. Which of the following node is considered as top of the stack?

First node.....confirm

Consider linked list is used to implement the Stack then which of the following node is considered as Top of the Stack ?

- ☐ A Middle Node
- ☐ B Last Node
- ☒ C First Node
- ☐ D Any Node

46. “+” is a _____ operator

Binary.....confirm

Precedence of Operators

There are five **binary operators**, called *addition, subtraction, multiplication, division* and *exponentiation*. We are aware of some other **binary operators**. For example, all **relational operators** are binary ones. There are some unary operators as well. These require only one operand e.g. $-$ and $+$. There are rules or order of execution of operators in Mathematics called precedence. Firstly, the *exponentiation* operation is

47. Convert the given infix form $12 + 60 - 23$ if expression in postfix form.

12 60 + 23 - confirm

Infix: 12+60-23

Infix to Postfix

Postfix: 1260+23-

48. Which of the following is the correct conversation from infix to postfix expression?

$A*B+C/(E-F)$

AB*CEF-/.....confirm

Infix: $A*B+C/(E-F)$

Infix to Postfix

Postfix: $AB*CEF-/+$

Step by

49. Stack.push(15) will push 15 on _____.

Top of stack....confirm

Stack.push(15) will push 15 on _____.

Top of the stack

50. In simple or singly linked list there is/are __ pointer/s in each node.

One.....confirm

In **simple or** singly linked list there is/are _____ pointer/s in each node.

1. One

51. Which of the following statement is correct for the variable “current—“?

Current = current - - confirm

current = **current -1** ;

52. Circular linked list solve the problem of ____ pointers/method of the doubly link list.

Add....confirm

Then we **solve the** problem by using the *add, start, length, next, remove* and *get* methods of the class *CList*.

53. Which one of the following is not a one-step method in list implementation using array?

Find()....confirm

The other methods `get()`, `length()` etc are **one-step** methods. They carry out their operation in one instruction. There is no need of any loop or other programming structures to perform the task. The `get()` method gets the value from the specified position just in one step. Similarly the `update()` method sets a value at the specific position just in **one-step**. The `length()` method returns the value of the size of the list. The other methods `back()`, `start()` and `end()` also perform their tasks only in one step.

54. Local variables of a function are stored in,

Stack.....confirm

One should be careful about transient objects that are stored by reference in data structures. We know that the **local variables of a function** are created on call stack. Those variables

55. A stack carries _____ behavior.

LIFO....confirm

Primarily, there are two operations of a stack; `push()` and `pop()`. **A stack carries *lifo* behavior i.e. last in, first out.**

56. The `--` is a decrement operator in C++ that decreases the value of the operand by _____.

One....confirm

The `--` is a decrement operator in C++ that **decreases the value** of the operand by one. The above statement can also be written as

57. In singly linked list which node will keep track of starting position of the list.

Next node.....confirm

In singly linked list which node **will keep** track of starting position of the list.

1. **Next Node**

58. Consider the linked list having data [6, 72, 35, 65, 25] stored in it. While current pointer is pointing to memory location having 72 stored in it. After calling `add(4)`

function on the following linked list current point will point to memory location having value?

4...confrim

Consider the linked list having data [6, 72, 35, 65, 25] stored in it. While current pointer is pointing to memory location having 72 stored in it. After calling add(4) function on the following linked list current point will point to memory location having value?

1. 72
2. 35
3. 4

59. Consider the linked list having data [6, 72, 35, 65, 25] stored in it. While current pointer is pointing to memory location having 72 stored in it. After calling remove () function on the following linked list current point will point to memory location having value?

6....confirm

Consider the linked list having data [6, 72, 35, 65, 25] stored in it. While current pointer is pointing to memory location having 72 stored in it. After calling remove() function on the following linked list current point will point to memory location having value?

Select the correct option

☐	65
☐	25
☒	6

Handwritten red annotations: A circle around the '6' option, a checkmark, and the text '6, 35'.

60. Consider the following expression:

$$P = A * B + C - D$$

In the above expression “-“ is a _____ operator

Binary

61. $Y = \&x[0];$

In the above statement, we get the address of the first location of the array x and store it in y. Here “y” is

Lvalue...confirm

```
y = &x[0];
```

In the above statement, we get the address of the first location of the array x and store it in y. As y is lvalue, so it can be used on left hand side. This means that the above statement is correct.

62. Josephus problem is resolved by the implementation of _____.

Circular linked list....confirm

Now we will see an example where circular link list is very useful. This is Josephus Problem. Consider there are 10 persons. They would like to choose a leader. The way

63. In node class one field is an integer data and other field will be ____.

Pointer to node....confirm

64. Clear() of a list is used to _____.

Clear a list(remove all elements)...confirm

clear();	Clear a list (remove all elements)
----------	------------------------------------

65. createList() of a list is used to _____.

Create a new list (presumably empty)....confirm

createList()	Create a new list (presumably empty)
--------------	--------------------------------------

66. copy() of a list is used to _____.

Set one list to be a copy of another....confirm

copy()	Set one list to be a copy of another
--------	--------------------------------------

67. insert(X,?) of a list is used to _____.

Insert element X at a particular position in the list....confirm

insert(X, ?)	Insert element X at a particular position in the list
--------------	---

68. remove(?) of a list is used to _____.

Remove element at some position in the list...confirm

<code>remove(?)</code>	Remove element at some position in the list
------------------------	---

69. `get(?)` of a list is used to _____.

Get element at a given position...confirm

<code>get(?)</code>	Get element at a given position
---------------------	---------------------------------

70. `update(X,?)` of a list is used to _____.

Replace the element at a given position with X...confirm

<code>update(X, ?)</code>	Replace the element at a given position with X
---------------------------	--

71. `find(X)` of a list is used to _____.

Determine if the element X is in the list...confirm

<code>find(X)</code>	Determine if the element X is in the list
----------------------	---

72. `length()` method of a list class is used to _____.

Returns the length of the list...confirm

<code>length()</code>	Returns the length of the list.
-----------------------	---------------------------------

73. First element of an int array can be manipulated by using the index zero as:

`Int[0];`

To manipulate the first element, we will use the index zero as `x[0]` and so on. The arrays look like in the memory as follows:

74. Which of the following is known as Last-In, First-Out or Lifo data structure?

Stack...confirm

Which one of the following is known as "Last-In, First-Out" or **LIFO Data Structure**?

1. Linked List
2. **Stack**

75. The computer memory can be thought of as a/an

Array confirm

The computer memory can be thought of as an array.

76. When add() operation of a linked list is called the following action is done.

A new node is made....confirm

We can create a new node in the following manner in the **add()** operation of the linked list with code in C++:

77. What will be the postfix expression of following infix expression?

$D + F * F / G$

$DEF * G / +$ confirm

What will be the postfix expression of following infix expression?
 $D + E * F / G$

1. $DE * F / G +$
2. $DE + F * G /$
3. $DEF * G / +$

78. To create a _____ we link the last node with the first node in the list.

Circularly-linked list....confirm

NULL pointers i.e. *prev* in the first node and *next* in the last node. A way around this potential hazard is to link the last node with the first node in the list to create a circularly-linked list.

79. Array cells are ____ in computer memory.

Contiguous

Array cells are
contiguous in
computer memory

80. Stack can be implemented by which of the following data structure?

Array or Linked list....confirm

Stack can be implemented using **array and linked list**.

81. Add(12) method of linked list class will;

Add 12 as value in linked list.....confirm

add(12) method of linked list class will:

Add 12 as value in linked list

82. Current pointer always points to the _____ of the linked list.

First node....confirm

element of the list is at index 1. Similarly in the linked list we need to know the starting point of the list. For this purpose, we have a pointer *head* that points to the first node of the list. If we don't use *head*, it will not be possible to know the starting position of the list. We also have a pointer *current* to point to the current node of the

83. _____ is the major factor to see the efficiency of a program.

Time.....confirm

level upto which these are efficient in terms of CPU's time consumption. Time is the major factor to see the efficiency of a program.

84. An efficient program executes faster and helps in _____ the usage of resources like memory and disk.

Minimize....confirm

with their use. Data structures help us to organize the data in the computer, resulting in more efficient programs. An efficient program executes faster and helps minimize the usage of resources like memory, disk. Computers are getting more powerful with the passage of time with the increase in CPU speed in GHz, availability of faster

85. A list is the collection of items of the _____.

Same type.....confirm

A list is the collection of items of the same type

86. Consider the following push operations of a Stack

Stack push (4);

Stack push (6);

Stack push (5);

Stack push (8);

If a user calls a pop() operation then which value will be returned?

8....confirm

87. When an executable program runs, it is loaded in the computer memory and becomes a _____.

Process.....confirm

memory and becomes a process,

88. Head node pointer always points to the _____ of the linked list?

Current node

direction. With the help of these pointers, you can move in the clockwise direction or anti-clockwise direction. Head node pointer will remain at its position. You don't need to change it. If there is a need to remove the node pointed by head node than you

89. The abstract data type refers to the basic mathematical concept that defines the

Data type....confirm

implemented with the use of a particular hardware or software data structure. The term abstract data type (ADT) refers to the basic mathematical concept that defines the data type. We have discussed four different implementations of the list data structure for various implementations of the list with the use of an array, the size of the

90. The next field in the last node in a singly-linked list is set to ____-.

NULL....confirm

Circularly-linked lists

Let's talk about circularly linked list. The next field in the last node in a singly-linked list is set to NULL. The same is the case in the doubly-linked list. Moving along a singly-linked list has to be done in a watchful manner. Doubly-linked lists have two

91. The postfix form of the expression $A+B*C$ and $(A+B)*C$ will be __

Different ...confirm

employ the infix form of expressions. Consider the infix expressions ' $A+B*C$ ' and ' $(A+B)*C$ '. The postfix versions are ' $ABC*+$ ' and ' $AB+C*$ ' respectively. The order

92. Each node in a singly linked list contains two fields, one field called data field while other field contains

Pointer to next node.....confirm

69. Each node in a singly linked list contains two fields, one field called data field while other field contains:

- a) Pointer to an integer
- b) Pointer to character
- c) Pointer to next node

93. On performing which task on the list implementing array has to visit the entire from start to end

Finding an element.....confirm

▪ find

The worst-case in find is that we may have to search the entire list. In find, we have to search some particular element say x . If found, the *currentNode* pointer is moved at that node. As there is no order in the list, we have to start search from the beginning of the list. We have to check the value of each node and compare it with x (value to be searched). If found, it returns true and points the *currentNode* pointer at that node otherwise returns false. Suppose that x is not in the list, in this case, we have to search the list from start to end and return false. This is the worst case scenario. Though time gets wasted, but we find the answer that x is not in the

94. Consider the linked list having data [6, 72, 35, 65, 25] stored in it. While current pointer is pointing to memory location having 35 stored in it. What will be the resultant linked list after calling add(18) function on the given linked list

6, 72, 35, 18, 65, 25.....confirm

95. In __ various cells of memory are not located continuously

Linked list....confirm

there is need of using linked memory in which the various cells of memory, are not located continuously. In this process, each cell of the memory not only contains the value of the element but also the information where the next element of the list is

96. Memory address is stored in
Pointer.....confirm

Memory address (Raise For Success) is stored in:

- Address operator
- Reference
- Pointer Correct

97. “- -” is a _____ operator
Unary.....confirm

Precedence of Operators

There are five binary operators, called addition, subtraction, multiplication, division and exponentiation. We are aware of some other binary operators. For example, all relational operators are binary ones. There are some unary operators as well. These require only one operand e.g. - and +. There are rules or order of execution of operators in Mathematics called precedence. Firstly, the exponentiation operation is

98. Which of the following is the limitation of the array?
Fix size.....confirm

Which of the following (Raise For Success) is the limitation of the array?

- Cannot store user define objects
- Cannot store more than 1000 elements
- Have to random access
- Fix size Correct

99. In singly linked list a node consists of two parts:
Object and pointer....confirm

In singly linked list a node consists of two parts:

1. Object and structure
2. Two pointers
3. Two objects
4. Object and pointer

100. Stack and Queue can be implemented using _____.

Singly Link List or doubly link list....confirm

requirements, we can change the list implementation. It can be implemented with the use of singly-link list or doubly link list. As long as the interface is same, a

101. _____ is a collection of elements arranged in a linear order.

Stack.....confirm

"Stack is a collection of **elements** arranged in a linear order"

102. Consider the following infix expression

$7/8 + 9$

If one converts the above expression into postfix, what would be the resultant expression?

78/9+ confirm

Infix: 7/8+9

Infix to Postfix

Postfix: 78/9+

Step by

103. Which of the following operation is used to get an element from the top of stack?

Push(x).....confirm

push(x)	Insert x as the top element of the stack
---------	--

104. Which one of the following operations returns and removes top value of the stack.

Pop ()....confirm

pop()	Remove the top element of the stack and return it.
-------	--

105. Which one of the following operations returns top value of the stack without removing it.

Top ().....confirm

top()	Return the top element without removing it from the stack.
-------	--

106. _____ method returns the top element of the stack without removing it.

Top()....confirm

top()	Return the top element without removing it from the stack.
-------	--

107. What will be the postfix expression of following infix expression?

$A * B / C + D - E$

$AB + CD / + E -$..confirm

What will be the postfix expression of following infix expression?

$A * B / C + D - E$

1. $AB / C * D + E -$
2. $AB * C / D + E -$
3. $AB * C + D E / -$
4. $AB + CD / + E -$

108. In the statement template <T>. T represents _____.

Generic data type.....confirm

function, we have written

```
template <class T>
Stack<T> :: Stack() ;
```

This means that we are writing a template function that uses T wherever a data type is written. As we have declared the Stack class as a template class, $<T>$ will be written with the class name before the access specifier (i.e. ::) while defining a method of the class. Then there is the implementation of the constructor in which we assign the value -1 to top and allocate memory by using the new operator for stack of size $MAXSTACKSIZE$ of type T and put its starting address in the pointer $nodes$. It is pertinent to note that we create an array of type T (i.e. a generic type). Similarly we define the destructor $\sim Stack$ which frees the memory by deleting the

109. The `_start` method of list will position the `currentNode` and `lastCurrentNode` at the start of the list

`Start ()`...confirm

have added the elements with data values 5, 13, 4, 8, 24, 48 and 12. Later, the `start()` method of list is called that will position the `currentNode` and `lastCurrentNode` at the start of the list. Now there is no need to worry about the implementation of the List.

110. Which of the following is the correct conversion of infix to postfix expression?

$ZB+DH-K/-$ confirm

Infix: $Z+B-(D-H)/K$

Infix to Postfix

Postfix: $ZB+DH-K/-$

111. Which of the following function don't belongs to the stack class?

Crash ().....confirm

which of the following function don't belongs to the stack class?

1. push()
2. pop()
3. crash()

112. Consider we have performed the following operations on a stack of size 5
Pust(10); Push(20); Push(30); Pop(); Pop(); Push(40);

1 or 20...both...confirm

Consider we have performed the following operations on **stack of** size 5.

```
Push(10);  
Push(20);  
Push(30);  
Pop();  
Pop();  
Push(40);
```

1. 10
2. 20
3. 40
4. 20

113. In singly linked list a node comprises of _____ field/s

Two.....confirm

A singly link list's node is divided into **two field**.

114. Each operator in a postfix expression refers to the previous __ operands.

One.....confirm

Each operator in a postfix expression refers to the previous _____ operand(s).

1. One

115. Which of the following signature of the constructor function for a stack is correct as a generic data type?

Template<stack T>;

116. Last node in circular linked list contains

No null pointer....confirm

21. Last node in circular linked list contains

a) At least one null pointer

b) No null pointer

117. Back() method of list class is used to

Move the current position backward one element....confirm

back()	Move the current position backward one element
--------	--

118. Next() method of list class is used to

Move the current position forward one element....confirm

next()	Move the current position forward one element
--------	---

119. Tail() method of list class is used to

Move the "current" pointer to the very last element....confirm

tail()	Moves the "current" pointer to the very last element
--------	--

120. Start() method of list class is used to

Moves the "current" pointer to the very first element....confirm

Functions	Description
start()	Moves the "current" pointer to the very first element

121. To search an element which method of the list searches the entire list from beginning to end?

Find().....confirm

To search an element (Raise For Success) which method of the list searches the entire list from beginning to end?

- end():
- find(): Correct

122. If we use array to implement list, then there is an issue that gives difficulty when:

We will increase its size.....confirm

If we use array to implement list, then there is an issue that it gives difficulty when:

1. We will access values randomly
2. We will remove data from it
3. We will increase its size

All Updated MCqs the yahn tak file bhe upload krdon ga tommarrow ya phir 7 jun ko...JAZAKALLAH...Yad rahy ye promise nhe ker rhaa hon ...zindagi or mout ka hmy nhe pata hota try kron ga apni...copy that ☺))

123. For a perfect binary tree of height 4, what will be the sum of highest of node.

26

$$S = (2^{h+1} - 1) - (h+1) = (2^{4+1} - 1) - (4+1) = (2^5 - 1) - (5) = 31 - 5 = 26$$

124. If Ahmed is cousin of Ali and Ali is cousin of Asad then Ahmed is also cousin of Asad.

Transitivity....confirm

If Ahmad is cousin of Ali and Ali is cousin of Asad then Ahmad is also cousin of Asad. This statement has the following property

1. Reflexivity
2. Symmetry
3. Transitivity

125. Suppose there are 100 elements in an equivalence class, so initially there will be 100 trees the collection of these trees is called.

Forest....confirm

Suppose there are 100 elements **in an** equivalence class, so initially there will be 100 trees. The collection of these trees is called

1. Cluster
2. Class
3. **Forest**

126. The percolate Down procedure will move the smaller value and bigger value.

Up,down....confirm

will move downwards. In this regard, percolate procedures may be of great help. The *percolateDown* procedure will move the smaller value **up and bigger value down**.

127. For a perfect binary tree of height h , having N nodes, the sum of height of nodes is

$N - h - 1$ confirm

the **sum of heights** of nodes will be equal to $N - h - 1$.

128. If we want to find median of 50 elements, then after applying **builtHeap** method, how manytime **deleteMin** method will be called

25....confirm

If we want to find median of 50 elements, then after applying **buildHeap** methd, how many times **deleteMin** method will be called?

1. 5
2. **25**

129. The Expression **if(!heap-> isEmpty())** check
Heap is not empty....confirm

The expression
if(! heap ->isFull())
check

1. Heap is not **empty**
2. **Heap is not full**

130. Which of the following heap method increase the value if key at position p by the amount delta?

decreaseKey(p, delta).....confirm

Which of the following **heap** method lowers the value of key at position 'p' by the amount 'delta'?

1. increaseKey(p,delta)
2. **decreaseKey(p,delta)**

131. Given the values are the array representation of heap; 12 23 26 31 34 44 56 64 78 100. What is the 5th smallest element in the given heap?

34.....confirm

given the values **are the array** representation of heap:
12 23 26 31 34 44 56 64 78 100
What is the 5th smallest element in the given heap?

1. 31
2. **34**

132. Given the values are the array representation of heap; 12 23 26 31 34 44 56 64 78 100. If we perform 4 deleteMin operations the last element deleted is?

31.....confirm

given the values **are the array** representation of heap:
12 23 26 31 34 44 56 64 78 100
If we perform 4 deleteMin operations, the last element deleted is _____.

1. **31**

133. If the elements are 3, 5, 1, 6 in which order they will be removed from the Queue one by one?

3 5 1 6

134. The _____ of a binary tree is the maximum level of its leaves (also called the depth).

Height...confirm

- The **height** of a binary tree is the maximum level of its leaves (also called the depth).

135. The _____ of a binary tree is the maximum level of any leaf in the tree.

Depth...confirm

The **depth** of a binary tree is the maximum level of any leaf in the tree.

136. Following is a keyword of C++ _____.

Delete....confirm

Following **is a keyword** of C++ _____ .

1. del
2. **delete**

137. Analyze the given code carefully and identify which type of binary tree traversal this is;

```
Void traversal ( treeNode<int>*treeNode){  
If(treeNode !=Null){  
Cout << *(treeNode->getinfo( ))<< “”;  
Traversal ( treeNode->getleft( ));  
Traversal ( treeNode->getRight( ));}
```

Postwar....confirm

Analyze the given code carefully and identify which type of binary tree **traversal** this is :void **traversal** (TreeNode<int>* treeNode){ if(treeNode != NULL) { **traversal** (treeNode->getLeft()); cout << *(treeNode->getInfo())<< " "; **traversal** (treeNode->getRight()); }}

Postwar

138. The _ of a node in a binary tree is defined as the height of its left subtree minus height of its right subtree.

Balance....confirm

The balance of a node in a binary tree is defined as the height of its left subtree minus height of its right subtree.

139. If p is pointing to a node in an existing tree then $\text{info}(p)$ means;
Returns the contents of node p ...confirm

We can define different operations on binary tree.
If p is pointing to a node in an existing tree, then

- $\text{left}(p)$ returns pointer to the left subtree
- $\text{right}(p)$ returns pointer to right subtree
- $\text{parent}(p)$ returns the father of p
- $\text{brother}(p)$ returns brother of p .
- $\text{info}(p)$ returns content of the node.

140. If p is pointing to a node in an existing tree then $\text{left}(p)$ means;
Returns pointer to the left subtree p confirm

We can define different operations on binary tree.
If p is pointing to a node in an existing tree, then

- $\text{left}(p)$ returns pointer to the left subtree

141. If p is pointing to a node in an existing tree then $\text{right}(p)$ means;
Returns pointer to right subtree....confirm

If p is pointing to a node in an existing tree, then

- $\text{left}(p)$ returns pointer to the left subtree
- $\text{right}(p)$ returns pointer to right subtree

142. If p is pointing to a node in an existing tree then $\text{parent}(p)$ means;
Returns the father of p confirm

If p is pointing to a node in an existing tree, then

- $\text{left}(p)$ returns pointer to the left subtree
- $\text{right}(p)$ returns pointer to right subtree
- $\text{parent}(p)$ returns the father of p

143. If p is pointing to a node in an existing tree then $\text{brother}(p)$ means;
Returns brother of p confirm

We can define different operations on binary trees.
If p is pointing to a node in an existing tree, then

- $\text{left}(p)$ returns pointer to the left subtree
- $\text{right}(p)$ returns pointer to right subtree
- $\text{parent}(p)$ returns the father of p
- $\text{brother}(p)$ returns brother of p

144. Every _ tree is a binary search tree.

AVL.....confirm

Every AVL tree is a binary search tree.

145. If a function has recursive call as the last statement it is known as ____.

Tail recursion.....confirm

has recursive call as the last statement, it is known as tail recursion.

146. During in-order traversal using recursive calls, if we found a node is NULL it means this node will satisfy following condition

It will not have left child.....confirm

During in-order traversal using recursive calls, if we found a node is NULL. It means this node will satisfy following condition.

Select correct option:

It will not have left child ok

147. In binary search tree deleting a node is easy if it is a _____ node

Root....confirm

In binary search tree deleting a node is easy if it is a..... node

Root

148. Which of the following operation returns the contents of a binary tree node?

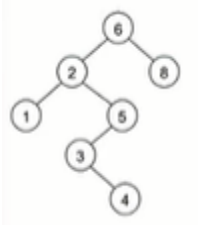
Info(p)....confirm

We can perform different operations on binary trees.

If p is pointing to a node in an existing tree, then

- $\text{left}(p)$ returns pointer to the left subtree
- $\text{right}(p)$ returns pointer to right subtree
- $\text{parent}(p)$ returns the father of p
- $\text{brother}(p)$ returns brother of p .
- $\text{info}(p)$ returns content of the node.

149. What will be level order of the given BST?



6,2,8,1,5,3,4

150. In a complete binary tree the number of nodes at level 5 are

31....confirm

no. of level in binary tree $5(5+1) + 1 = 31$

151. One should be careful about transient _____ that are stored by reference in data structures.

Objects....confirm

One should be careful about transient objects that are stored by reference in data structures.

152. In a tree we link the nodes in such a way that it _____ a linear structure.

Does not remain....confirm

allocate memory, link and set pointers to it. It is not much difficult process we link the nodes in such a way that it does not remain a linear structure.

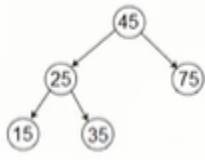
100,000, we have 1 million or 10 million or even 1 billion numbers and we

153. The function calls are made with the help of _____.

Stack....confirm

that function calls are made with the help of stacks.

154. What will be the postorder of the given tree?



15,35,25,75,45

155. A binary tree is said to be a _____ binary tree if every non-leaf node in a binary tree has non-empty left and right subtrees.

Strictly....confirm

There is a version of the binary tree, called strictly binary tree. A binary tree is said to be a strictly binary tree if every non-leaf node in a binary tree has non-empty left and right subtrees.

156. In which of the following tree for each node the value of root node is greater than left child and smaller than right child?

Complete binary tree...confirm

268. In which of the following tree for each node, the value of root node is greater than left child and smaller than right child?

a) Binary search Tree

b) Full Binary Tree

c) Complete Binary Tree

157. Two common models of simulation are _____ and _____.

Time-based, event-based....confirm

Let's discuss little bit about the simulation models. Two common models of simulation are time-based simulation and event-based simulation. In time-based

158. Generally there is/are _ case(s) to delete a node from BST.

3.....confirm

There are ____ cases for deleting a node from binary search tree.

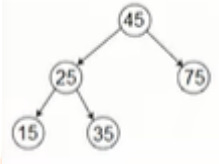
- a. 1
- b. 2
- c. 3

159. When a function calls another function the parameters and return address of the calling function are put in ____.

Stack...confirm

We know that function calls are made with the help of stacks. When a function calls some other function, the parameters and return address of the function is put in a stack. The local variables of the function are also located at the stack and the control

160. What will be the inorder traversal of the given tree?



15,25,35,45,75

161. ~BinarySearchTree() is a ____
Constructor.....confirm

~BinarySearchTree() is a ____.

1. Constructor

162. We can calculate the ____ of a subtree by counting its levels from the bottom.

Height....confirm

We can calculate the height of a subtree by counting its levels from the bottom.

163. If there is a strictly complete binary tree of depth 3 the total number of nodes in it will be?

7....confirm

If **there is a** strictly complete binary tree of depth 3 the total number of nodes it will be

7

164. In which of the following function signatures the value of variable "num" cannot be changed in function body?

Int cube (int num).....confirm

In which of the following function signatures, the **value of variable** "num" cannot be changed in function body?

1. int cube(int num)

165. A _____ model attempts to model a real-world phenomenon.

Simulation.....confirm

Out of the numerous uses of the queues, one of the most useful is *simulation*. A simulation program attempts **to model a** real-world phenomenon. Many popular video

166. The process of getting the value of a variable using pointers is called
Dereferencing.....confirm

The process of **getting** the value of a variable using a pointer is called

Dereferencing

167. When a function calling itself is called as _____.

(Recursive or recursion) both.....confirm

A function that calls itself is said to be **recursive**.

168. A _____ is a tree in which every level except possibly the last is completely filled

Complete binary tree....confirm

A is a tree in which every level except possibly the last is completely filled

Complete binary tree

169. _____ objects (objects accessed by pointers) are called anonymous objects.

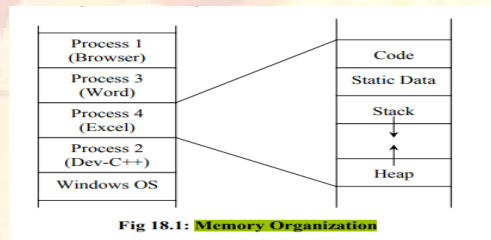
Nameless objects....confirm

starting addresses of those objects to *c1* and *c2* pointers. Nameless objects (objects accessed by pointers) are called **anonymous** objects.

Here *c1* and *c2* are pointers to the objects not the actual objects themselves as it was

170. In the perspective of memory organization each process is divided into _____ sections.

4.....confirm



171. Analyze the given code carefully and choose the statement which will be used if we want to set 7 as the value of root?

```
Int x [] = {14,15,4,9,7,18,4,5,16,4,20,17};
```

```
TreeNode * root = new TreeNode( );
```

```
Root->setinfo( &x[4]);
```

172. In tree, the search operation is _____ as compared to the linked list.

Very fast....confirm

Thus in a tree, the search is very fast as compared to the linked list.

173. Level-order traversal for binary search tree can be implemented _____.

Only through recursive method confirm

Level-order traversal (Raise For Success) for Binary Search Tree can be implemented _____.

- Only through recursive method Correct

174. _____ is an area in computer memory that is used to allocate memory dynamically.

Heap....confirm

Heap is an area in computer memory that is allocated dynamically.

175. Suppose we want to create the right child of a node 'p' and set the value x in it. Which of the following statement is correct for this operation?

setRight (x,p)....confirm

Suppose we want to create the right child of 'p' and set the value x in it. which of the following statement is correct in this operation

Set Right (x,p)

176. In level-order traversal for binary Search Tree at each level we visit the nodes in _____ order

Left-to-right.....confirm

traversal procedures discussed previously. In level-order traversal, we visit the nodes at each level before proceeding to the next level. At each level, we visit the nodes in a left-to-right order. The traversal methods discussed earlier are inorder, preorder and

177. Which of the following function(s) is/are used for function call by value?

```
Void func1(int& num) {
```

```
Num++;}
```

```
Void func2(int num){
```

```
Num++;}
```

Both func1 and func2

178. In a program a reference variable with name 'x' can be declared as

`Int &x;....confirm`

55. In a program a reference variable, say x, can be declared as **a) int &x;**

179. Suppose the elements are 3,5,1,6, if we want to enqueue '8' then dequeue all the elements what will be the order of elements?

8,3,5,1,6

After **dequeue()** is called once

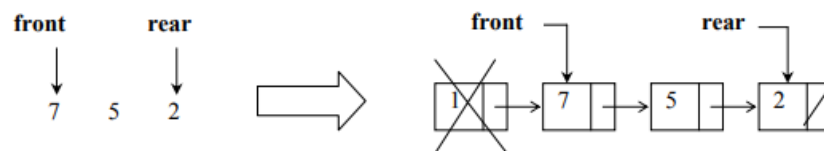


Fig 4. Removal of one element from queue using **dequeue()**

180. Which of the following data structure is linear type?

(List, Queue, Stack)....All options

49. Which of the following data structure **is linear** type?

- a) Stack
- d) List
- c) Queue
- d) All of the above**

181. Analyze the given code carefully and identify which type of binary tree traversal this is;

```
Void traversal ( treeNode<int>*treeNode){  
If(treeNode !=Null){  
Cout << *(treeNode->getinfo( ))<< “”;  
Traversal ( treeNode->getleft( ));  
Traversal ( treeNode->getRight( ));}}
```

Inorder

182. Doubly Linked list always has one Null pointer.

False....confirm

implemented using list, there will be 1000 extra pointer variables. Implementing this stack is implemented using 'singly-linked' list. Otherwise, for doubly linked list, this overhead is also doubled as two pointer variables are stored within each node in that case.

