#1 Explained Report Bookmark

What is the worst case time complexity of linear search algorithm?

- A
  O(log n)
- B
  O(n)
- C
  O(1)
- D
  O(n^2)

Correct Answer :B

Explanation

A linear search scans one item at a time, without jumping to any item .

1. The worst case complexity is O(n), sometimes known an O(n) search
2. Time taken to search elements keep increasing as the number of elements are increased.

#2 Explained Report Bookmark

A pivot element to partition unsorted list is used in

- A
  Quick Sort
- B
  Merge Sort
- C
  Selection Sort

- D
  Insertion Sort

Correct Answer :A

Explanation

Quick sort work on divide and conquer algorithm. In this we choose a pivot element which divides the list into two parts. The first part elements are smaller than pivot and the other side elements are larger.

#3 Explained Report Bookmark

Name the node which has been generated but none of its children nodes have been generated in the state-space tree of the backtracking method.

- A
  Dead Node
- B
  Live Node
- C
  E-Node
- D
  State Node

Correct Answer :B

Explanation

Because in state space tree organization live node is a node which has been generated but none of its children nodes have been generated, but are being generated.

#4 Explained Report Bookmark

How many nodes are there in a full state space tree with n = 6?

- A
  65
- B
  64
- C
  63
- D
  32

Correct Answer :C

Explanation

State space tree comes under backtracking algorithm. And it is tree constructed from all possible states of the problem as node which is connect via state transitions from some initial state as called root to some terminal state

And to find nodes in state space tree we use a formula:- 2^n-1,

As this is (2^(n))-1 = (2^6)-1=64-1= 63

#5Explained Report Bookmark

This algorithm scans the list by swapping the entries whenever pair of adjacent keys are out of the desired order.

- A
  Inserion Sort
- B
  Bubble Sort
- C
  Shell Short
- D
  Quick Short

Correct Answer :B

Explanation

In bubble sort we repeatedly swap the adjacent element, if the order of elements are not correct . this is the simple algorithm which scan the list by swapping in each iteration

#6 Explained Report Bookmark

From the following chose the one which belongs to the algorithm paradigm other than to which others from the following belongs to.

- A
  Minimum and Maximum Problem
- B
  Knapsack Problem
- C
  Selection Problem
- D
  Merge Sort

Correct Answer :B

Explanation

Here , the Minimum and maximum Problem in algorithm use to find the maximum and minimum value in an array. And the selection sort algorithm sort an array by repeatedly finding the minimum element from the unsorted list and put it at the beginning position

But the knapsack problem is the optimization problem . In this we have a set of items with its specific weights and values. In this we aim to get as much value into the knapsack as possible but the total weight should be less or equal to the knapsack weight which is given in the problem.

#7Not Explained Report Bookmark

To calculate c(i, j)’s, w(i, j)’s and r(i, j)’s; the OBST algorithm in the worst case takes the following time.

null

- A
  O(log n)
- B
  O(n4)
- C
  O(n3)
- D
  O(n log n)

Correct Answer :C

No Explanation Available

#8Not Explained Report Bookmark

Greedy job scheduling with deadlines algorithms’ complexity is defined as

null

- A
  O(N)
- B
  Ω(n log n)
- C
  O(n log n)
- D
  Ω(n2 log n)

Correct Answer :A

No Explanation Available

#9 Explained Report Bookmark

What is time required to insert an element in a stack with linked implementation ?

- A
  (log2n)
- B
  (n)
- C
  (n log2n)
- D
  O(1)

Correct Answer :D

Explanation

In stack, there are some basic operation which are

Push , Pop and the stack operation only perform on the top of the stack

When we push the element in the stack . It take O(1) time and when we pop the element , it remove from the top of the stack so it also take O(1) time

#10 Explained Report Bookmark

When is linear queue said to be empty ?

- A
  Front=rear
- B
  Front !=rear
- C
  Front=rear-1

- **D**
  Front=rear+1

Correct Answer :A

Explanation

when the linear queue is empty the front and rear should be -1

#11 Explained Report Bookmark

Elements can be retrieved by index in …………….?

- **A**
  linked lists
- **B**
  linear arrays
- **C**
  both of above
- **D**
  none of above

Correct Answer :B

Explanation

Array is a collection of similar types of data items which are stored in contiguous memory. And in an array we retrieve element by its index and the indexing start from 0

0 1 2

9 6 3

#12 Explained Report Bookmark

Efficiency of an algorithm is measured by

- A
  Time and Capacity complexity
- B
  Time and Space complexity
- C
  Speed and Space complexity
- D
  Speed and Capacity complexity

Correct Answer :B

Explanation

Algorithm Efficiency. Time efficiency - a measure of amount of time for an algorithm to execute. Space efficiency - a measure of the amount of memory needed for an algorithm to execute. Asymptotic dominance - comparison of cost functions when n is large

#13 Explained Report Bookmark

Which sorting algorithm is the slowest algorithm for large number of data?

null

- A
  Quick sort
- B
  Heap sort

- C
  Bubble sort
- D
  Shell sort

Correct Answer :C

Explanation

Quick sort, Heap sort and Shell sort all have best case time complexity as O(nlogn) and Bubble sort has time complexity of O(n2).

So, Bubble sort is slowest.

#14 Explained Report Bookmark

Retrieval operation is fastest in which data structure

- A
  Sorting
- B
  Hashing
- C
  Indexing
- D
  both A and C

Correct Answer :B

Explanation

hashing is the fastest way to retrieve . In hashing larger values converted into small keys with the help of hash function . and the hashing is generally used in almost situation

#15 Explained Report Bookmark

State True or False about array and linked list

I. Retrieval of element will be faster in array than link list?

II. Search operation in both array and link list are leaner.

- A
  True, False
- B
  False, True
- C
  False, False
- D
  True, True

Correct Answer :D

Explanation

is true because in array retrieval is done by index number which is O(1) complexibility

#16 Explained Report Bookmark

In which data structure memory is contiguous

- A
  Link List
- B
  Array
- C
  None
- D
  Both

Correct Answer :B

Explanation

Array store in contiguous memory but the linked list is scattered in the memory connect with the help of pointer

#17 Explained Report Bookmark

True statements about Stack and Queue are

I. Stack and Queue both are linear data structures

II. Stack is non- linear data structure.

III. Stack is LIFO

IV. Queue is FIFO

- A
  I and II only
- B
  I and III only
- C
  I, III and IV
- D
  All are correct.

Correct Answer :C

Explanation

In the above statement 1st , 3rd and 4th statement is correct because

Both stack and queue are linear data structure and stack follow LIFO (last in first out) algo. And queue follow (FIFO) first in first out

#18 Explained Report Bookmark

A Binary Tree can have

- **A**
  Can have 1 children
- **B**
  Can have 0 children
- **C**
  Can have 2 children
- **D**
  All

Correct Answer :D

Explanation

Binary tree is which each node can have utmost 2 children (it means each node has either 0 child or 1 child or 2 child)

#19 Explained Report Bookmark

Height of a binary tree is

- **A**
  MAX(Height of left Subtree, Height of right subtree)+1
- **B**
  MAX(Height of left Subtree, Height of right subtree)
- **C**
  MAX(Height of left Subtree, Height of right subtree)-1
- **D**
  None

Correct Answer :A

Explanation

For example, height of an empty tree is 0 and height of tree with only one node is 1. The idea is to traverse the tree in post-order fashion and calculate the height of left and right subtree. The height of the subtree rooted at any node will be equal to maximum height of its left and right subtree plus one.

#20 Explained Report Bookmark

Postfix expression for (A+B) *(C+D) is

null

- A
  A B C * + D +
- B
  A B + C D + *
- C
  ABCD++*
- D
  None

Correct Answer :B

Explanation

In postfix form two operands come together then operator. For example, in A+B, A and B are two operands and + is one operator and A+B is in infix form. Hence, in postfix of A+B will be AB+.

Let’s consider the question above

(A+B) *(C+D)//this expression is in infix form

First get A and B operand together then operator +. Similarly, for C and D

(AB+) * (CD+)

now consider AB+ and CD+ as two operands. Take them together then * operator.

(AB+) (CD+) *

Remove brackets.

AB+CD+*

So, AB+CD+* is the postfix expression of (A+B) *(C+D) infix expression.

#21 Explained Report Bookmark

True statements about AVL tree are

- **A**
  It is a binary search tree.
- **B**
  Left node and right node differs in height by at most 1 unit
- **C**
  Worst case time complexity is O(log2n)
- **D**
  All

Correct Answer :D

Explanation

AVL tree is a simple Binary Search Tree (BST) with some advantage where the difference between heights of left and right subtrees cannot be more than one for all nodes

It is also called self balancing or balance BST

#22 Explained Report Bookmark

Which one of the following is an application of Stack Data Structure?

- A
  Managing function calls
- B
  The stock span problem
- C
  Arithmetic expression evaluation
- D
  All of the above

Correct Answer :D

Explanation

The function call stack (just as the *call stack* or *the stack*) is responsible for maintaining the local variables and parameters during function execution. We use stack in recursion (a process in which a function call itself)

And when we execute the arithmetic expression we generally use stack which follow some steps to evaluate the expression

#23 Explained Report Bookmark

Which of the following is true about the linked list implementation of stack?

null

- A
  In `push` operation, if new nodes are inserted at the beginning of linked list, then in `pop` operation, nodes must be removed from end.
- B
  In `push` operation, if new nodes are inserted at the end, then in `pop` operation, nodes must be removed from the beginning.
- C
  Both of the above

- D
  None of the above

Correct Answer :D

Explanation

None of the Above. To keep the Last In First Out (LIFO) order, a stack can be implemented using linked list in two ways: a) In push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from beginning. b) In push operation, if new nodes are inserted at the end of linked list, then in pop operation, nodes must be removed from end.

#24 Explained Report Bookmark

Given below is a pseudocode that uses a stack. What will be the output for input "Studytonight" for the given pseudocode:

```
declare a stack of characters;



while (there are more characters in the word to read)

{

    read a character;

    push the character on the stack;

}

while (the stack is not empty)
```

```
{

    pop a character off the stack;

    write the character to the screen;

}
```

- A
  StudytonightStudytonight
- B
  thginotydutS
- C
  Studytonight
- D
  thginotydutSthginotydutS

Correct Answer :B

Explanation

thginotydutS. Since the stack data structure follows LIFO order.

When we `pop()` items from stack, they are popped in reverse order of their insertion (or `push()`)

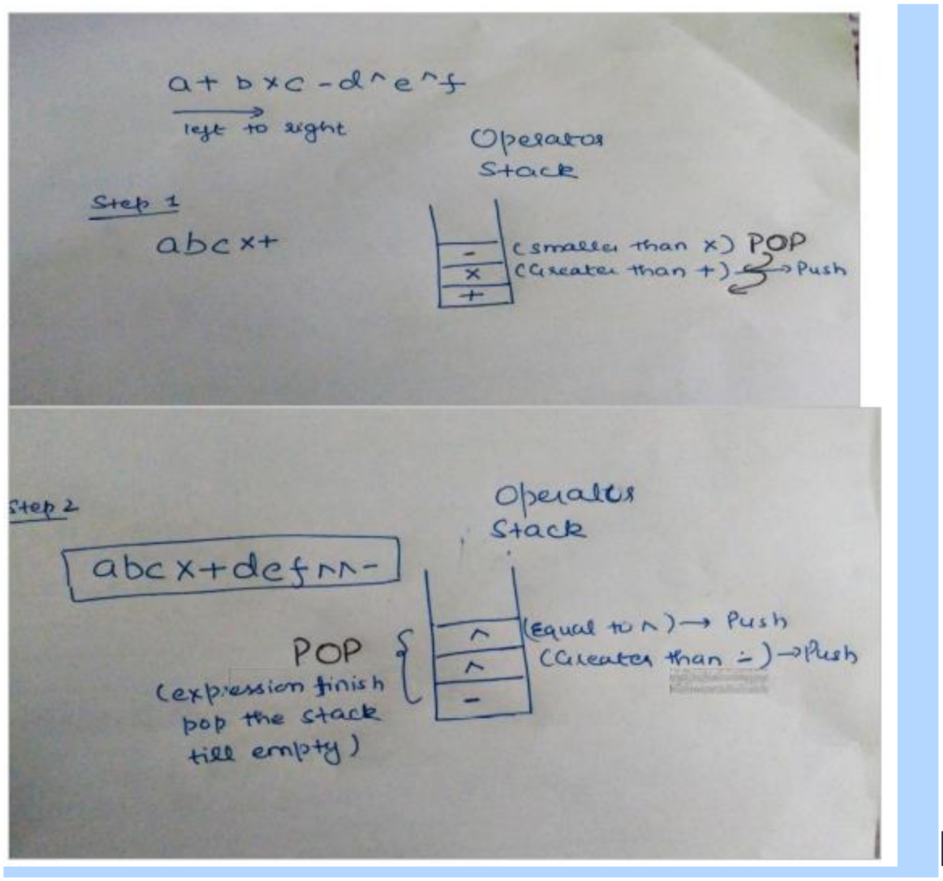
#25 Explained Report Bookmark

Assume that the operators +, -, * are left associative and ^ is right associative. The order of precedence (from highest to lowest) is ^, *, +, -. The postfix expression corresponding to the infix expression a + b x c - d ^ e ^ f will be ?

- A
  `abcx+def^^-`
- B
  abcx+de^f^-

- **C**
  ab+cxd-e^f^
- **D**
  `-+axbc^^def`

Correct Answer :A



Explanation

#26 Explained Report Bookmark

To evaluate an expression without any embedded function calls:

null

- **A**
  One stack is enough

- B
  Two stacks are needed
- C
  As many stacks as the height of the expression tree are needed
- D
  A Turing machine is needed in the general case

Correct Answer :A

Explanation

One stack is enough. Any expression can be converted into Postfix or Prefix form. Prefix and postfix evaluation can be done using a single stack.

#27 Explained Report Bookmark

The result of evaluating the postfix expression 10 5 + 60 6 / * 8 - will be?

- A
  284
- B
  213
- C
  142
- D
  72

Correct Answer :C

Explanation

It scan the operand and operator one at a time and put it in the stack and when we get operator it will evaluate the last two operand and push the result in the stack

10 5 + 60 6 / * 8 -

1)10 (push it in stack)

10

2) 5 (push it again in stack)

5

10

3) + (it will add last two number and push the result in stack which is 10+5 = 15)

15

4)60 (push it in stack)

60

15

5)6 (push it in stack)

6

60

15

6) / (60/6= 10)

10

15

7) * (10 * 15 = 150)

150

8) 8 (push it in stack)

8

150

9)- (150 - 8 = 142)

142

Here we get our final answer which is 142

#28 **Explained** **Report** **Bookmark**

Suppose a stack is to be implemented using a linked list instead of an array. What would be the effect on the time complexity of the push and pop operations of the stack implemented using linked list (Assuming stack is implemented efficiently)?

null

- **A**
  O(1) for insertion and O(n) for deletion
- **B**
  O(1) for insertion and O(1) for deletion
- **C**
  O(n) for insertion and O(1) for deletion
- **D**
  O(n) for insertion and O(n) for deletion

Correct Answer :B

Explanation

Stack can be implemented using link list having `O(1)` bounds for both insertion as well as deletion by inserting and deleting the element from the beginning of the list.

#29 Explained Report Bookmark

Consider n elements that are equally distributed in k stacks. In each stack, elements are arranged in ascending order (minimum value is at the top in each of the stack and then increasing downwards). Given a queue of size n in which we have to put all the n elements in increasing order. What will be the time complexity of the best known algorithm?

null

- A
  O(nlogk)
- B
  O(nk)
- C
  $O(n^2)$
- D
  $O(k^2)$

Correct Answer :A

Explanation

It can be done in `nlog k`, by creating a min heap of size `k` and adding all the top-elements of all the stacks. After extracting the min, add the next element from the stack from which we have got our 1st minimum. Time Complexity = `O(k)` (For Creating Heap of size `k`) + `(n-k)log k` (Insertions into the heap).

#30 Explained Report Bookmark

Which of the following statements is true:

i) First in First out types of computations are efficiently supported by STACKS.

ii) Implementing LISTS on linked lists is more efficient than implementing LISTS on an array for almost all the basic LIST operations.
iii) Implementing QUEUES on a circular array is more efficient than implementing QUEUES on a linear array with two indices.
iv) Last in First out type of computations are efficiently supported by QUEUES

- A
  (ii) and (iii) are true
- B
  (i) and (ii) are true
- C
  (iii) and (iv) are true
- D
  (ii) and (iv) are true

Correct Answer :A

Explanation

In linked list we do not need to declare its length because of its dynamic nature that's the reason to implement list on linked list is more efficient

#31 Explained Report Bookmark

A priority queue Q is used to implement a stack S that stores characters. PUSH(C) is implemented as INSERT(Q, C, K) where K is an appropriate integer key chosen by the implementation. POP is implemented as DELETEMIN(Q). For a sequence of operations, the keys chosen will be in:

- A
  Non-increasing order
- B
  Non-decreasing order
- C
  strictly increasing order
- D
  strictly decreasing order

Correct Answer :D

Explanation

We are implementing a STACK using Priority Queue. Note that Stack implementation is always last in first out (LIFO) order. As given 'POP is implemented as DELETEMIN(Q)' that means Stack returns minimum element always. So, we need to implement PUSH(C) using INSERT(Q, C, K) where K is key chosen from strictly-decreasing order(only this order will ensure stack will return minimum element when we POP an element). That is the answer, option (D) is true.

#32 Explained Report Bookmark

A function f defined on stack of integers satisfies the following properties: f(∅) = 0 and f (push (S, i)) = max (f(S), 0) + i for all stacks S and integers i. If a stack S contains the integers 2, -3, 2, -1, 2 in order from bottom to top, what is f(S)?

null

- A
  6
- B
  4
- C
  3

- D
  2

Correct Answer :C

Explanation

f(S) = 0, max(f(S), 0) = 0, i = 2 f(S)new = max(f(S), 0) + i = 0 + 2 = 2

f(S) = 2, max(f(S), 0) = 2, i = -3 f(S)new = max(f(S), 0) + i = 2 - 3 = -1

f(S) = -1, max(f(S), 0) = 0, i = 2 f(S)new = max(f(S), 0) + i = 0 + 2 = 2

f(S) = 2, max(f(S), 0) = 2, i = -1 f(S)new = max(f(S), 0) + i = 2 - 1 = 1

f(S) = 1, max(f(S), 0) = 1, i = 2 f(S)new = max(f(S), 0) + i = 1 + 2 = 3

#33Explained Report Bookmark

Following is C like pseudocode for a function that takes a number as an argument, and uses a stack S to do the processing. What will the below function do in general?

```
void fun(int n)
{
    Stack S;  // Say it creates an empty stack S
    while (n > 0)
    {
        // This line pushes the value of n%2 to stack S
        push(&S, n%2);
        n = n/2;
    }
    // Run while Stack S is not empty
    while (!isEmpty(&S))
    {
        printf("%d ", pop(&S)); // pop an element from S and print
it
    }
}
```

- **A**
  Prints binary representation of `n` in reverse order
- **B**
  Prints binary representation of `n`
- **C**
  Prints the value of `Log n`
- **D**
  Prints the value of `Log n` in reverse order

Correct Answer :B

Explanation

suppose in this program we pass 4 and it store in n

Now n=4 and then it create an empty stack and check the while loop condition

1st iteration , n>0 => 4>0(true) it enter inside the loop and push the result of n%2 in the stack which is 4%2 = 0

Now , n = 4/2 , n=2

2nd iteration, 2>0 (true) again it push the result of n % 2 which is again 0

And , n =2/2, n= 1

3rd iteration , 1> 0 (true) then n % 2 -> 1%2 which is 1 pushed in the stack and n= ½ =0.5 and it consider as 0

4th iteration , 0>0(false), now execute the next while loop and stack look like

1

0

0

Now in second loop , we just run until the stack empty and pop all the elements one by one in each iteration and print on the console

First we pop 1 (in stack operation perform from one end which is top of the stack), then 0 and again 0

And it look like 100 which is the binary representation of 4 which we passed in function

#34 Explained Report Bookmark

A single array A[1..MAXSIZE] is used to implement two stacks. The two stacks grow from opposite ends of the array. Variables top1 and top2 (topl < top 2) point to the location of the topmost element in each of the stacks. If the space is to be used efficiently, the condition for "stack full" will be:

- A
  `(top1 = MAXSIZE/2)` and `(top2 = MAXSIZE/2+1)`
- B
  top1 + top2 = MAXSIZE
- C
  `(top1 = MAXSIZE/2)` or `(top2 = MAXSIZE)`
- D
  top1 = top2 -1

Correct Answer :D

Explanation

If we want to use space efficiently then size of the any stack can not be more than MAXSIZE/2. Both stacks will grow from both ends and if any of the stack `top` reaches near to the other top then stacks are full. So the condition will be top1 = top2 -1 (given that top1 < top2)

#35 Explained Report Bookmark

Which one of the following is an application of Queue Data Structure?

- A
  When a resource is shared among multiple consumers.
- B
  When data is transferred asynchronously (data not necessarily received at same rate as sent) between two processes.
- C
  Load Balancing.
- D
  All of the above

Correct Answer :D

Explanation

Queue is used in cpu scheduling and disk scheduling , in print spooling, documents are loaded into a buffer and then the printer pulls them off the buffer . Spooling also lets you place a number of print jobs on a queue.

When data is transferred asynchronously between two processes.Queue is used for synchronization. Examples : IO Buffers, pipes, file IO

Also In real life, Call Center phone systems will use Queues

#36 Explained Report Bookmark

How many stacks are needed to implement a queue. Consider the situation where no other data structure like arrays, linked list is available to you.

- A
  1
- B
  2
- C
  3
- D
  4

Correct Answer :B

Explanation

queue can be implemented using two stacks.

#37 Explained Report Bookmark

A Priority Queue can be efficiently implemented using which of the following data structures?

- A
  Array
- B
  Linked List
- C
  Heap Data Structures like Binary Heap, Fibonacci Heap
- D
  None of the above

Correct Answer :C

Explanation

Heap Data Structures like Binary Heap, Fibonacci Heap.

#38 Explained Report Bookmark

Which of the following is true about linked list implementation of queue?

- A
  In `push` operation, if new nodes are inserted at the beginning of linked list, then in `pop` operation, nodes must be removed from end.
- B
  In `push` operation, if new nodes are inserted at the end, then in `pop` operation, nodes must be removed from the beginning.
- C
  Both of the above
- D
  None of the above

Correct Answer :C

Explanation

Both of the above. To keep the First In First Out order, a queue can be implemented using linked list in any of the given two ways.

In this we can perform operation on both end , if we insert the element from one end then the other end is use for deleting the element

Suppose if we are inserting the element from the end then we will delete the element from beginning

In the queue we call the both end as rear and front , generally we use front for deleting the element and rear for inserting , but it depends on the user .

#39 Explained Report Bookmark

Suppose a Circular queue of capacity (n - 1) elements is implemented with an array of n elements. Assume that the insertion and deletion operation are carried out using REAR and FRONT as array index variables, respectively. Initially, REAR = FRONT = 0. The conditions to detect, queue is full and queue is empty will be:

- A
  Full: `(REAR+1) mod n == FRONT`, Empty: `REAR == FRONT`
- B
  Full: `(REAR+1) mod n == FRONT`, Empty: `(FRONT+1) mod n == REAR`
- C
  Full: `REAR == FRONT`, Empty: `(REAR+1) mod n == FRONT`
- D
  Full: `(FRONT+1) mod n == REAR`, Empty: `REAR == FRONT`

Correct Answer :A

Explanation

#40 Explained Report Bookmark

A Priority Queue is implemented as a Max-Heap. Initially, it has 5 elements. The level-order traversal of the heap is given below: 10, 8, 5, 3, 2. Two new elements 1 and 7 are inserted in the heap in that order. The level-order traversal of the heap after the insertion of the elements will be:

- A
  10, 8, 7, 5, 3, 2, 1
- B
  10, 8, 7, 2, 3, 1, 5
- C
  10, 8, 7, 1, 2, 3, 5
- D
  10, 8, 7, 3, 2, 1, 5

Correct Answer :D

Explanation

```
Initially heap has 10, 8, 5, 3, 2
    10
   /  \
  8    5
 / \
3   2

After insertion of 1
     10
   /   \
  8     5
 / \   /
3   2 1
No need to heapify as 5 is greater than 1.


After insertion of 7
     10
   /   \
  8     5
 / \   / \
3   2 1   7
Heapify 5 as 7 is greater than 5
     10
   /   \
  8     7
 / \   / \
3   2 1   5
No need to heapify any further as 10 is


greater than 7
```

#41 Explained Report Bookmark

Suppose a stack implementation supports an operation REVERSE, which reverses the order of elements on the stack, in addition to the PUSH and POP operations. Which one of the following statements is TRUE with respect to this modified stack?

null

- A
  A queue cannot be implemented using this stack.
- B
  A queue can be implemented where `ENQUEUE` will be implemented using a single operation and `DEQUEUE` will require two stack operations.
- C
  A queue can be implemented where `ENQUEUE` will be a combination of 3 stack operations and `DEQUEUE` will require a single operation.
- D
  A queue can be implemented where both `ENQUEUE` and `DEQUEUE` will require a single stack operation.

Correct Answer :C

Explanation

To `DEQUEUE` an item, simply `POP`.

To `ENQUEUE` an item, we will have to use the following 3 operations: 1) `REVERSE`, 2) `PUSH` 3) `REVERSE`.

#42 Explained Report Bookmark

A queue is implemented using an array such that ENQUEUE and DEQUEUE operations are performed efficiently. Which one of the following statements is CORRECT if the array is circular. (n refers to the number of items in the queue)?

- A
  Both operations can be performed in O(1) time

- B
  At most one operation can be performed in O(1) time but the worst case time for the other operation will be Ω(n)
- C
  At most one operation can be performed in O(1) time but the worst case time for the other operation will be Ω(n)
- D
  Worst case time complexity for both operations will be Ω(log n)

Correct Answer :A

Explanation

Using circular array, both operations can be performed in O(1) time.

#43 Explained Report Bookmark

What is recurrence for worst case of QuickSort and what is the time complexity in Worst case?

null

- A
  Recurrence is `T(n) = T(n-2) + O(n)` and time complexity is `O(n2)`
- B
  Recurrence is `T(n) = T(n-1) + O(n)` and time complexity is `O(n2)`
- C
  Recurrence is `T(n) = 2T(n/2) + O(n)` and time complexity is `O(nlogn)`
- D
  Recurrence is `T(n) = T(n/10) + T(9n/10) + O(n)` and time complexity is `O(nlogn)`

Correct Answer :B

Explanation

The worst case of QuickSort occurs when the selected pivot is always one of the corner elements in the sorted array. In worst case, QuickSort recursively calls one subproblem with size 0 and other subproblem with size (n-1). So recurrence is T(n) = T(n-1) + T(0) + O(n)

#44 Explained Report Bookmark

Suppose we have a O(n) time algorithm that finds median of an unsorted array. Now consider a QuickSort implementation where we first find median using the above algorithm, then use median as pivot. What will be the worst case time complexity of this modified QuickSort

null

- A
  `O(n2*logn)`
- B
  O(n2)
- C
  O(n*logn*logn)
- D
  `O(n*logn)`

Correct Answer :D

Explanation

O(nlogn). If we use median as a pivot element, then the recurrence for all cases becomes T(n) = 2T(n/2) + O(n) The above recurrence can be solved using Master Method. It falls in case 2 of master method.

#45 Explained Report Bookmark

What is the worst case time complexity of Insertion sort where position of the data to be inserted is calculated using Binary search?

- A
  N

- B
  NlogN
- C
  N^2
- D
  $N(\log N)^2$

Correct Answer :C

Explanation

Applying binary search to calculate the position of the data to be inserted doesn’t reduce the time complexity of insertion sort. This is because insertion of a data at an appropriate position involves two steps:

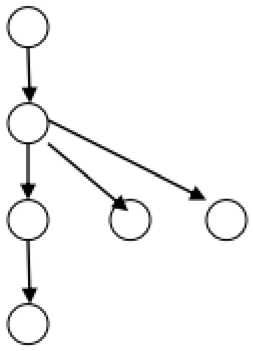
1. Calculate the position.

2. Shift the data from the position calculated in step #1 one step right to create a gap where the data will be inserted.

Using binary search reduces the time complexity in step #1 from O(N) to O(logN). But, the time complexity in step #2 still remains O(N). So, overall complexity remains O(N^2).

#46 Explained Report Bookmark

What is the maximum height of stack required to process the given directed graph using DFS (Depth first search) ?



- A
  5
- B
  3
- C
  4
- D
  6

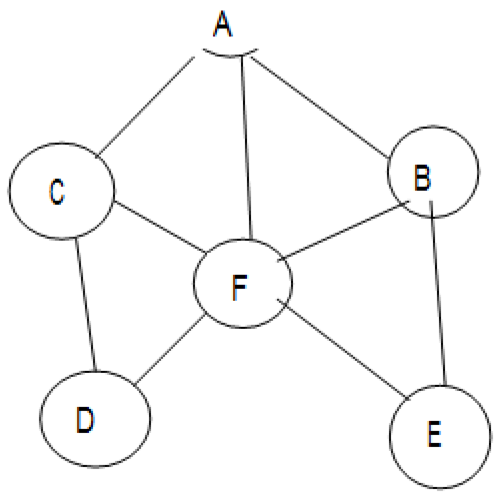
Correct Answer :C

Explanation

In order to process above graph for DFS , a stack of minimum height 4 is required.

#47 Explained Report Bookmark

Which of the following BFS (Breadth First Search) sequence is valid for the given graph



- **A**
  ACFDEB
- **B**
  AFBCDE
- **C**
  DCBEF
- **D**
  DBFEC

Correct Answer :B

Explanation

Consider the starting node 'A' and apply BFS. Then the valid traversing sequence will be AFBCDE only

#48 Explained Report Bookmark

What is the maximum height of queue (To keep track of un-explored nodes) required to process a connected Graph G1 which contains 'N' node using BFS algorithm?

- **A**
  (N-1)/2

- B
  N-1
- C
  (N/2)-1
- D
  N

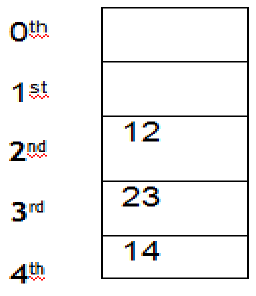
Correct Answer :B

Explanation

When we apply BFS on any connected graph having 'N' nodes then all the nodes except the starting node is enqued onto the queue. Therefore answer is N-1

#49 Explained Report Bookmark

What will be the probability that the upcoming element will be hashed to 0th cell using linear probing: Hash function : h(k) = k mod 10 Where K denotes the key to be inserted.



- A
  4/5
- B
  1/5
- C
  2/5
- D
  3/5

Correct Answer :A

Explanation

All the possible keys which are supposed to be mapped to the cells 2nd,3rd,4th and 0th will be hashed to 0th cell only ,therefore probability will increase to 4/5.It is a typical case of clustering

#50 Explained Report Bookmark

Which of the following permutations can be obtained in the output (In the same order) using a stack .Assuming the Input is the sequence A,B,C,D,E in that order.

- A
  CDEAB
- B
  CDEBA
- C
  AEBCD
- D
  EDCAB

Correct Answer :B

Explanation

Please find the below steps :

Push A

Push B

Push C

Pop C

Push D

Pop D

Push E

Pop E

Pop B

Pop A

hence the sequence will be CDEBA

#51 Explained Report Bookmark

Consider a single linked list with start node pointed by 'head' pointer. Suppose there is a middle node 'X' which is pointed by a pointer 'ptr'. What is the time complexity of the best known algorithm to delete the node X .

null

- A
  O(n)
- B
  O(1)
- C
  O(n^2)
- D
  None of the above

Correct Answer :B

Explanation

Ptr->data =ptr->next->data; Ptr->next = ptr->next->next;

#52 Explained Report Bookmark

Suppose a new data type 'X' is created to hold integers. The maximum number of bits allocated to the this X is 5. Consider the number to be represented in 2's compliment form then what will be the range of the values which can be assigned to X.

null

- A
  -16 to +15
- B
  -16 to +16
- C
  .-15 to +15
- D
  -15 to +16

Correct Answer :A

Explanation

When a number is represented in 2's compliment form then the range of the number becomes [-2n− 1 , +2n− 1-1] . Put n=5 here to get the desired range.

#53 Explained Report Bookmark

Consider a hash table with 50 slots .Collision is resolved using chaining. What is the probability that there will be No collision in any of the 50 slots

null

- **A**
  $50! / (50^{50})$
- **B**
  $(50 \times 49 \times 48) / (50^{50})$
- **C**
  1/2
- **D**
  $(1/50^{50})$

Correct Answer :A

Explanation

The above condition can be done easily by using Combination concept as follows:(50□X 48□ X …………………. X 1□) / (50□X 50□X 50□X 50□…………….50 times)1 1 1 1 1 1 1 1

#54 Explained Report Bookmark

What is the maximum number of movements required in order to get the correct location of an element in insertion sort?

null

- **A**
  N
- **B**
  N-1
- **C**
  N-2
- **D**
  N-3

Correct Answer :B

Explanation

Consider a case where Nth element has shift to first location then all N-1 elements will have to shift one place. Therefore total number of movements is N-1.

#55 Explained Report Bookmark

How many queues are needed to implement priority queues?

- A
  3
- B
  2
- C
  4
- D
  5

Correct Answer :B

Explanation

Queue 1 -> for storing data

Queue 2 -> for storing priority

#56 Explained Report Bookmark

Which of the following sorting algorithm has least best case time complexity?

- A
  Radix sort
- B
  Insertion sort
- C
  Quick sort

- D
  Shell sort

Correct Answer :B

Explanation

Best case for insertion sort : When elements are already arranged.

#57 Explained Report Bookmark

The small o notation in asymptotic evaluation represents?

- A
  Average case
- B
  loose lower bound
- C
  loose upper bound
- D
  tight lower bound

Correct Answer :C

Explanation

The little o notation is one of them. Little o notation is used to describe an upper bound that cannot be tight. In other words, loose upper bound of f(n). ... We can say that the function f(n) is o(g(n)) if for any real positive constant c, there exists an integer constant n0 ≤ 1 such that f(n) > 0.

#58 Explained Report Bookmark

There are K nodes in a binary search tree. What will be the maximum number of edges in this tree?

null

- A
  K
- B
  K+1
- C
  2K-1
- D
  K-1

Correct Answer :D

Explanation

For any tree having n nodes can contain n-1 edges only

#59 Explained Report Bookmark

Which of the following condition represents Overflow for a queue?

- A
  Front = Rear
- B
  Front = Rear -1
- C
  Front =Rear + 1
- D
  none of these

Correct Answer :C

Explanation

Click Here for more details

#60Explained Report Bookmark

which of the following is a non-linear data structure?

null

- A
  linked list
- B
  Array
- C
  Graph
- D
  None of these

Correct Answer :C

Explanation

Array and linked list are the example of linear data structures. Graph , trees etc are the example of nonlinear DS

#61Explained Report Bookmark

which of the following problems can not be implemented by a stack?

- A
  Creation of process mix by long term scheduler
- B
  Balancing parenthesis
- C
  valuation of post fix notation
- D
  Infix to postfix conversion

Correct Answer :A

Explanation

Creation of process mix by long term scheduler cannot be implemented by a stack.

#62 Explained Report Bookmark

Quick sort uses which of the following paradigm

null

- A
  Greedy
- B
  Dynamic programming
- C
  Back tracking
- D
  Divide & conquer

Correct Answer :D

Explanation

Quick sort uses Divide and conquer for its implementation .

#63 Explained Report Bookmark

What is the result of following postfix expression? 2 4 1 * - 3 -

null

- A
  -2

- B
  -1
- C
  -5
- D
  4

Correct Answer :C

Explanation

We make use of stack in order to evaluate postfix expression.

Push 2

Push 4

Push 1

Pop 4 ,PoP 1

Multiply 4 * 1 =4

Push 4

Pop 4 ,Pop2

Subtract 2 -4 =-2

Push -2

Push 3

Pop -2 ,Pop 3

Subtract -2 -3 = -5

#64 Explained Report Bookmark

Evaluate the following postfix expression?

3 9 + 6 4 / * 3 5 * *

- A
  160
- B
  180
- C
  190
- D
  200

Correct Answer :B

Explanation

Push 3

Push 9

Pop 9 ,Pop 3

Add 9 + 3 =12

Push 12

Push 6

Push 4

Pop 4 ,Pop 6

Divide 6 /4 =1

Push 1

Pop 1, Pop 12

Multiply 12 * 1 =12

Push 12

Push 3

Push 5

Pop 3 ,Pop 5

Multiply 3 * 5= 15

Push 15

Pop 15 ,pop 12

Multiply 12 * 15 = 180

#65 Explained Report Bookmark

Which sorting algorithm is best suited when elements are present in increasing order?

- A
  Selection Sort
- B
  Merge sort
- C
  Heap sort
- D
  Insertion sort

Correct Answer :D

Explanation

Insertion sort takes O(n) time which is best suited when elements are arranged in increasing order.

#66 Explained Report Bookmark

Which data structure is best suited for implementing huffman's encoding algorithm?

null

- A
  Array
- B
  Linked List
- C
  Graphs
- D
  Heaps

Correct Answer :D

Explanation

In every iteration of huffman's encoding we need two nodes with minimum frequency.This task can be done by min heaps in minimum time using extract_min() finction [O[log n]] .

#67 Explained Report Bookmark

In quick sort algorithm what is the time complexity of partition function when last element is selected as pivot element?

null

- A
  O(n)
- B
  O(log n)
- C
  O(n log n)
- D
  O(n^2)

Correct Answer :A

Explanation

Time complexity of partition function in quick sort is O(n).

#68 Explained Report Bookmark

Which of the following is out of place sorting algorithm?

null

- A
  Insertion sort
- B
  Bubble sort
- C
  Selection sort

- D
  Merge sort

Correct Answer :D

Explanation

An algorithm which takes extra space for its processing is known as out of place algorithm. In merge sort we require O(n) extra space for its MERGE procedure.

#69 Explained Report Bookmark

Which of the following sorting algorithms does not have a worst case running time of O(n2)?

- A
  Insertion sort
- B
  Merge sort
- C
  Quick sort
- D
  Bubble sort

Correct Answer :B

Explanation

Insertion sort, bubble sort and quick sort algorithms have O(n^2) worst-case time complexity.

#70 Explained Report Bookmark

The average time required to perform a successful sequential search for an element in an array A(1 : n) is given by

- A
  (n + 1)/2
- B
  $\log_2 n$
- C
  n(n + 1)/2
- D
  n^2

Correct Answer :A

Explanation

This is because in average case half of the array may be searched to find a required element.

#71 Explained Report Bookmark

A search begins the search with the element that is located in the middle of the array

- A
  serial
- B
  random
- C
  parallel
- D
  binary

Correct Answer :D

Explanation

In binary search, mid element is found first and search for the required element starts from there.

#72 Explained Report Bookmark

Average successful search time for sequential search on 'n' items is

null

- A
  n/2
- B
  (n-1)/2
- C
  (n+1)/2
- D
  None of these

Correct Answer :C

Explanation

If search key matches the very first item, with one comparison we can terminate. If it is second, two comparisons, etc.

Average =[n (n + 1)] / 2

#73 Not Explained Report Bookmark

Suppose DATA array contains 1000000 elements. Using the binary search algorithm, one requires only about n comparisons to find the location of an item in the DATA array, then n is

null

- A
  60

- B
  45
- C
  20
- D
  15

Correct Answer :C

No Explanation Available

#74 Explained Report Bookmark

Which of the following statements is used in the binary search algorithm to halve the array ?

- A
  middle Sub = (start Sub + stop Sub)/2;
- B
  middle Sub = start Sub + stop Sub/2;
- C
  middle Sub = middle Sub/2;
- D
  middle Sub = (stop Sub - start Sub)/2;

Correct Answer :A

Explanation

mid element is found in the binary search algorithm with the help of the formula given below:

Mid element= (lower bound + upper bound)/2

#75 Explained Report Bookmark

The number of swappings needed to sort the numbers 8, 22, 7, 9, 31, 5, 13 in ascending order, using bubble sort is

- A
  10
- B
  11
- C
  12
- D
  14

Correct Answer :A

Explanation

1 : 8, 7, 9, 22, 5, 13, 31 = 4 swaps

2 : 7, 8, 9, 5, 13, 22, 31 = 3 swaps

3 : 7, 8, 5, 9, 13, 22, 31 = 1 swap

4 : 7, 5, 8, 9, 13, 22, 31 = 1 swap

5 : 5, 7, 8, 9, 13, 22, 31 = 1 swap

Total 10 swaps are required to sort the array.

#76 Explained Report Bookmark

Which of the following is a collection of items into which items can be inserted arbitrarity and from which only the smallest item can be removed?

- A
  Descending priority queue
- B
  Ascending priority queue
- C
  Fifo queue
- D
  None of these

Correct Answer :B

Explanation

An ascending priority queue is a collection of items into which Items can be inserted arbitrarily and from which only the smallest item can be removed. The priority queue is a data structure in which the intrinsic ordering of the elements does determine the result of its basic operations.

#77 Explained Report Bookmark

Queues serve major role in

- A
  Simulation of recursion
- B
  Simulation of arbitrary linked list
- C
  Simulation of limited resource allocation
- D
  All of above

Correct Answer :C

Explanation

Simulation of recursion uses stack data structure. Simulation of arbitrary linked lists uses linked lists. Simulation of resource allocation uses queue as first entered data needs to be given first priority during resource allocation. Simulation of heap sort uses heap data structure.

#78 Explained Report Bookmark

Which of the following data structure may give overflow error,even though the current number of element in it is less than its size?

- A
  Simple queue
- B
  Circular queue
- C
  Priority Queues
- D
  None of these

Correct Answer :A

Explanation

Lets take an example we have 5 elements {10,20,30,40,50 } to be added .

we have two elements rear and front .

Initially front is set to -1 and rear is 0

Now when we are adding an element it will increase rear pointer by 1 and if we remove one element it will increase front by 1 .

So if we have added 5 elements in the Queue rear will be at 4th index i.e. size of queue -1 and if we are removing one element front will be increased to 0 from -1 and the element is assumed to be deleted then even after deletion of 1 element (i.e. 10) we have a space of 1 element but Queue will give the message of overflow because rear is still at 4 .

#79 Explained Report Bookmark

Number of "ADD" and "REMOVE" operations required to access n/2th elements of a queue of "n" elements so that the original queue remain the same after the access is (take help of another queue.)

- A
  4*n
- B
  8*n
- C
  4*n-
- D
  8*n-1

Correct Answer :A

Explanation

basically it takes 2n push pop operations in order to fetch the element. But since we are supposed to perform an operation and keep the queue as it is we must perform more 2n operation on the queue again. Which will make it 4n. Let's say you have 6 elements in a queue. So it will take 3 remove and 3 add operations in order to fetch the middle element. Then again to keep the queue as it is you must remove all remaining elements from the queue apart from those which were added earlier that are the last 3 elements, To do this you need to perform 3 remove and 3 add operations. Then again when you are done working on the middle element, you are supposed to keep it in the place where it belonged.e. n/2th position. So in order to do that again perform 3 remove, 3 add, insert element, 2remove and 2 add. Thus resulting it takes 4n ADD and REMOVE operations to access the queue's middle element.

#80 Explained Report Bookmark

The initial configuration of a queue is a, b, c, d, ('a' is in the front end). To get the configuration d, c, b, a, one needs a minimum of

- A
  2 deletions and 3 additions
- B
  3 deletion and 2 additions
- C
  3 deletions and 3 additions
- D
  3 deletions and 4 additions

Correct Answer :C

Explanation

Queue is a first in first out data structure and in this problem using this, 3 deletions and 3 additions are required.

#81 Explained Report Bookmark

Queue can be used to implement

- A
  radix sort
- B
  quick sort
- C
  recursion
- D
  depth first search

Correct Answer :A

Explanation

Queue can be used to implement radix sort.

#82 Explained Report Bookmark

Which of the following statements is false ?

- A
  Every tree is a bipartite graph
- B
  A tree contains a cycle
- C
  A tree with n nodes contains n-1 edges
- D
  A tree is a connected graph

Correct Answer :B

Explanation

Since a tree contains no cycles at all, it is bipartite. Every connected graph with only countably many vertices admits a normal spanning tree

#83 Explained Report Bookmark

A binary tree in which if all its levels except possibly the last, have the maximum number of nodes and all the nodes at the last level appear as far left as possible, is called

- A
  Full binary tree
- B
  Binary Search Tree

- C
  Threaded tree
- D
  Complete binary tree

Correct Answer :D

Explanation

A complete binary tree is a binary tree in which every level, except possibly the last, is completely filled, and all nodes are as far left as possible.

#84 Explained Report Bookmark

A complete binary tree with the property that the value at each node is at least as large as the values at its children is called

- A
  binary search tree
- B
  Binary Tree
- C
  Completely balanced tree
- D
  Heap

Correct Answer :D

Explanation

Heap is a special case of balanced binary tree data structure where the root-node key is compared with its children

Generally , heap can be two types: 1st Max heap and 2nd Min heap

1. Max-Heap: In a Max-Heap the key present at the root node must be greatest among the keys present at all of its children.
2. Min-Heap: In a Min-Heap the key present at the root node must be minimum among the keys present at all of its children

#85 Explained Report Bookmark

A full binary tree with n leaves contains

- A
  n nodes
- B
  log2n nodes
- C
  2n - 1 nodes
- D
  2n+1 nodes

Correct Answer :C

Explanation

a full binary tree with N leaves contains 2n - 1 nodes

#86 Explained Report Bookmark

A full binary tree with n non-leaf nodes contains

- A
  logn nodes
- B
  n + 1 nodes
- C
  2n-1 nodes
- D
  2n + 1 nodes

Correct Answer :D

Explanation

A full binary tree with n non leaf nodes contain 2n+1 nodes.

#87 Explained Report Bookmark

A 3-ary tree in which every internal node has exactly 3 children. The number of leaf nodes in such a tree with 6 internal nodes will be

- A
  10
- B
  09
- C
  12
- D
  13

Correct Answer :D

Explanation

An K-Ary tree is a tree in which nodes can have at most K children.

In this we use a formula to find the leaf node in k-ary tree(3-ary) , here we have 6 internal node

Formula : l = (k - 1) * n +1, here k =3 and n = 6

l= (3 - 1) * 6 + 1 ,

l= 2 * 6 +1,

l= 12 + 1 = 13

#88 Explained Report Bookmark

A complete binary tree of level 5 has how many nodes ?

- A
  15
- B
  25
- C
  63
- D
  30

Correct Answer :C

Explanation

formula used: (2^(h+1))-1

max number of nodes = 2^(h+1)-1 = 2^6-1 =63.

Click Here For More Detail

#89 Explained Report Bookmark

Traversing a binary tree first root and then left and right subtrees called _______traversal.

- A
  postorder
- B
  preorder
- C
  inorder
- D
  none of these

Correct Answer :B

Explanation

preorder has the property of traversing root first then left child and finally right child.

#90 Explained Report Bookmark

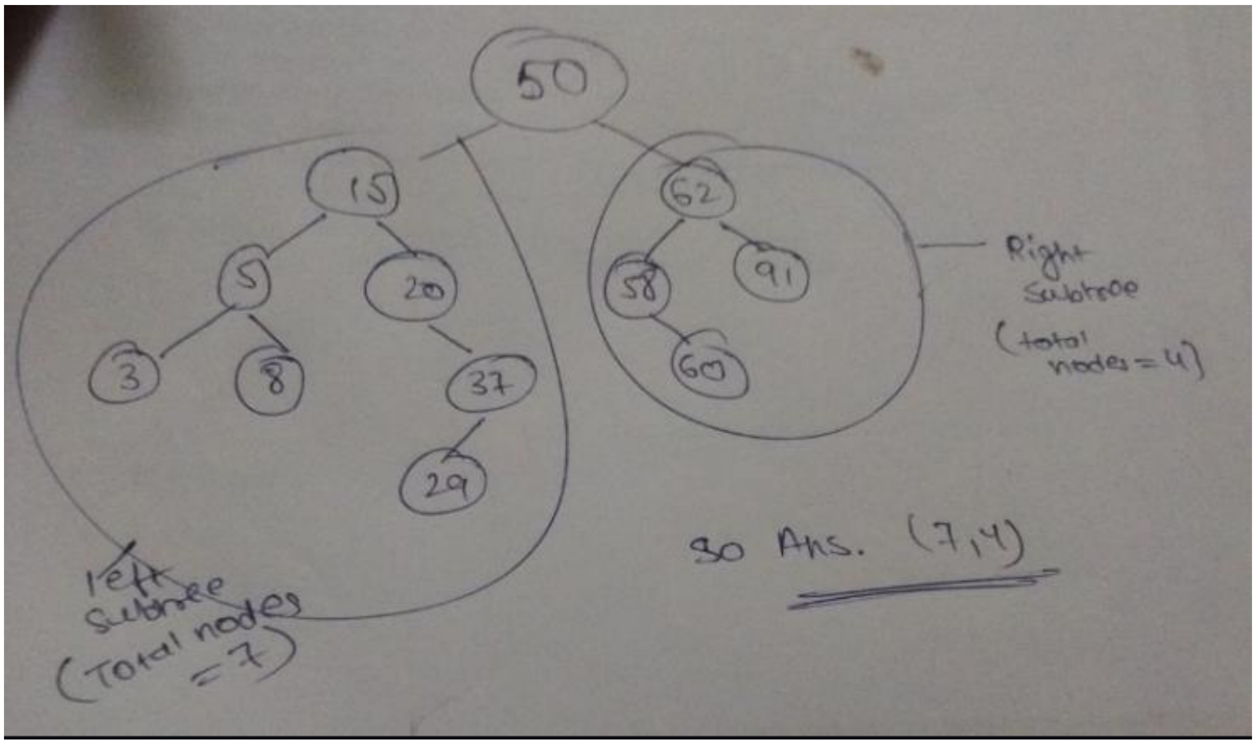
A binary tree is generated by inserting in order the following integers: 50, 15, 62, 5, 20,58, 91, 3,8,37, 60, 24 The number of nodes in the left and right of the root respectively is

- A
  (4,7)
- B
  (7,4)
- C
  (6,3)
- D
  (3,6)

Correct Answer :B

Explanation

In the binary tree we start creating a tree with root node 50 and insert the other number one by one . when we get a number which is less than 50 place at left subtree and if number is greater than 50 placed it at right subtree. Here ,


50
15
5
20
3
8
37
29
62
58
91
60
Right Subtree (total nodes = 4)
left subtree (Total nodes = 7)
So Ans. (7,4)

#91 Explained Report Bookmark

The number of possible ordered trees with 3 nodes A, B, C is

- A
  16
- B
  12
- C
  6
- D
  10

Correct Answer :B

Explanation

At 1st split the problem into two parts -

A) Tree with 3 nodes i.e root node has two children

B) Tree with 2 nodes i.e root has one child.

Now consider case 'A'. Here look root node is fixed and remaining two child nodes are able to interchange their positions.

As example

Let A is root node and B,C are child nodes. Now B can be left child & C can be right child. Alternatively B can be right and C can be left. So they can interchange their positions in 2! ways. Similarly If we consider B as root then we get another 2! and in case of C as root got another 2!. So from case 'A' we got total 2!+2!+2! = 6 number of distinct ordered trees. Now lets go to case 'B'. Here it's all about root and exactly one child i.e just 2 nodes.

So here the node combinations are A-B, B-C, C-A & like the case 'A' for A-B we got 2!, for B-C: 2! & for C-A: 2!.

So in case 'B' also total 2!+2!+2! = 6 number of distinct ordered trees.

Now we have come to the final part. If you look carefully, then you find that case 'A' & 'B' are non-overlapping cases as we consider distinct suit of conditions for each case respectively.

So now simply add the results we got from case 'A' & 'B' i.e 6+6 = 12.

Therefore, we can conclude now by our final result as 12. :) :)

#92 **Not Explained** **Report** **Bookmark**

Which of the following expressions accesses the (i,j)th entry of an (m x n) matrix stored in column major form?

null

- A
  n x (i -1) + j
- B
  m x (n-j) + j
- C
  m x(j -1) + i
- D
  n x(m-i) + j

Correct Answer :C

No Explanation Available

#93 Explained Report Bookmark

Which of the following is useful in traversing a given graph by breadth first search?

- A
  stacks
- B
  set
- C
  List
- D
  Queue

Correct Answer :D

Explanation

BFS uses a Queue data structure

DFS uses a Stack data structure

More detail Click Here

#94 Explained Report Bookmark

The recurrence relation that arises in relation with the complexity of binary search is

- A
  T(n)=T(n/2)+k, where k is constant
- B
  T(n)=2T(n/2)+k, where k is constant
- C
  T(n)=T(n/2)+log(n)
- D
  T(n)=T(n/2)+n

Correct Answer :A

Explanation

Binary Search is a linear searching algorithm and takes O(log n) when array is

sorted.

T(n) = T(n / 2) + k , where k is constant produces a complexity of O(log n)

#95 Explained Report Bookmark

The running time T(n), where `n' is the input size of a recursive algorithm is given as

followsT(n)=c+T(n-1),if n>1 d, if n≤ 1 The order of this algorithm is

- A
  n2
- B
  n
- C
  n3
- D
  nn

Correct Answer :B

Explanation

By recursively applying this relation we finally arrive at T (n-1)= c (n-I)+ T(1)

= (n-I)+d

= so order is n

#96 Explained Report Bookmark

The concept of order(Big O) is important because

- A
  it can be used to decide the best algorithm that solves a given problem
- B
  It is the lower bound of the growth rate of the algorithm
- C
  It determines the minimum size of a problem that can be solved in a given system, in a given amount of time
- D
  It is the average bound of the growth rate of the algorithm

Correct Answer :A

Explanation

The *Big O* notation defines an upper *bound* of an *algorithm*, it *bounds* a function only from above.Big O notation is used in Computer Science to describe the performance or complexity of an algorithm. Big O specifically describes the worst-case scenario, and can be used to describe the execution time required or the space used (e.g. in memory or on disk) by an algorithm.

#97 Explained Report Bookmark

The concept of order(Big O) is important because

- A
  it can not be used to decide the best algorithm that solves a given problem
- B
  It determines the maximum size of a problem that can be solved in a given system, in a given amount of time
- C
  It is the lower bound of the growth rate of the algorithm
- D
  It is the average bound of the growth rate of the algorithm

Correct Answer :B

Explanation

let problem A of size is n.

Big O says time taken by best algorithm that solve every case(best, avarage, worst) of problem A completely.

in general Big O is "The Best algorithm of worst case input".

it also determines the max size of problem that can be solved in given amount of time.

#98 Explained Report Bookmark

The time complexity of an algorithm

T(n), where n is the input size is given by

T(n)=T(n-1)+/n, if n>1 =1

otherwise. The order of the algorithm is

- A
  log n
- B
  n
- C
  n^2
- D
  n^n

Correct Answer :A

Explanation

$T(n)=T(n-1)+\frac{1}{n}----------(1)$$T(n)=T(n-1)+\frac{1}{n}----------(1)$

$T(n-1)=T(n-2)+\frac{1}{n-1}--------(2)$$T(n-1)=T(n-2)+\frac{1}{n-1}--------(2)$

$T(n-2)=T(n-3)+\frac{1}{n-2}--------(3)$$T(n-2)=T(n-3)+\frac{1}{n-2}--------(3)$

put T(n-1) from equation 2 to eq. 1 then result will be put T(n-1) from equation 2 to eq. 1 then result will be

$T(n)=T(n-2)+\frac{1}{n}+\frac{1}{n-1}--------(4)$$T(n)=T(n-2)+\frac{1}{n}+\frac{1}{n-1}--------(4)$

put T(n-2) from equation 3 to eq. 4 then result will be put T(n-2) from equation 3 to eq. 4 then result will be

T(n)=T(n− 3)+1n+1n− 1+1n− 2T(n)=T(n− 3)+1n+1n− 1+1n− 2

Similarly we can write-->Similarly we can write-->

T(n)=T(n− 4)+1n− 3+1n− 2+1n− 1+1nT(n)=T(n− 4)+1n− 3+1n− 2+1n− 1+1n

⟹T(n)=T(0)+∑i=1n1i=0+Hn,⟹T(n)=T(0)+∑i=1n1i=0+Hn,

where HnHn are the harmonic numbers.

Hn=1+12+13+14+ ⋯+1n=log(n)Hn=1+12+13+14+ ⋯+1n=log(n)

T(n)=θ(log(n))

#99 Explained Report Bookmark

The running time of an algorithm is given by

T(n)=T(n-1)+T(n-2)-T(n-3),

if n>3= n

otherwiseThe order of this algorithm is

- A
  n
- B
  log n
- C
  n^n

- **D**
  n2

Correct Answer :A

Explanation

From the definition, we can see that:

$T(1)=1T(1)=1$

$T(2)=2T(2)=2$

$T(3)=3T(3)=3$

$T(4)=T(3)+T(2)-T(1)=3+2-1=4T(4)=T(3)+T(2)-T(1)=3+2-1=4$

$T(5)=T(4)+T(3)-T(2)=4+3-2=5T(5)=T(4)+T(3)-T(2)=4+3-2=5$

$T(6)=T(5)+T(4)-T(3)=5+4-3=6T(6)=T(5)+T(4)-T(3)=5+4-3=6$

We can observe the pattern. It looks like $T(n)=nT(n)=n$, but we need to prove it.

Proof by strong induction:

Hypothesis: $T(n)=nT(n)=n$

Base Case: $T(1)T(1)$ to $T(6)T(6)$ have been shown above.

Assume, for the sake of induction $T(x), \forall x \in [1, n-1]$$T(x), \forall x \in [1, n-1]$

Then,

$T(n) = T(n-1) + T(n-2) - T(n-3)$

$= (n-1) + (n-2) - (n-3)$

$= n$

#100 Explained Report Bookmark

If n=4,then the value of O(log n) is

- **A**
  1
- **B**
  2
- **C**
  4
- **D**
  8

Correct Answer :B

Explanation

Here, we are doing O(n *n) which becomes $O(n^2)$ (O of n squared). This is known as quadratic time which means that every time the number of elements increases.

#101 Explained Report Bookmark

If n=4,then the value of O(n2) is

- A
  4
- B
  16
- C
  64
- D
  512

Correct Answer :B

Explanation

n=4

O(n^2) = 4^2 = 16

Note : if it is O(2n) then ans is 4 only instead of 8. Because asymptotic notations ignore constants.

#102 Explained Report Bookmark

The average time complexity of insertion sort is

- A
  O(n2)

- B
  O(n)
- C
  O(1)
- D
  O(log n)

Correct Answer :A

Explanation

This algorithm is not suitable for large data sets as its average and worst case complexity are of O(n2), where n is the number of items.

#103 Explained Report Bookmark

The running time of an algorithm is given by

T(n)=T(n-1)+T(n-2)-T(n-3),

if n>3= n

otherwise What should be the relation between T(1), T(2) and T(3)

so that its order is constant.

- A
  T(1)=T(2)=T(3)
- B
  T(1)+T(3)=2T(2)
- C
  T(1)-T(3)=T(2)
- D
  T(1)+T(2)=T(3)

Correct Answer :A

Explanation

As per definition if n=n>3 else n=n

$T(n)=Tn-1)+T(n-2)-T(n-3).$

If the order of the algorithm for which above equation is applicable for the time complexity, is a constant then,

$T(n)=T(n-1).$

$T(n)=T(n)+T(n-2)-T(n-3)$

$T(n-2)=T(n-3)$

Thus T(n)=T(n)=T(n) until n<3

So we can conclude that A is the right answer

#104 Explained Report Bookmark

The order of the algorithm that finds a given Boolean function of `n' variables , produces a is

- A
  constant
- B
  linear
- C
  non-linear
- D
  exponential

Correct Answer :D

Explanation

In the worst case it has to check all the 2^n possible input combinations, which is exponential.

#105 Not Explained Report Bookmark

If n=16, then the value of O(n log n) is

null

- A
  16
- B
  32
- C
  64
- D
  128

Correct Answer :C

No Explanation Available

#106 Explained Report Bookmark

How many memory management functions are there in C

- A
  4
- B
  3
- C
  2

- D
  1

Correct Answer :C

Explanation

C provides 2 methods of allocating memory to the variables and programs. They are static and dynamic memory allocations. In static memory allocation, memory is allocated at the time of compilation and will be same throughout the program.

#107 Explained Report Bookmark

If n= 8, then the value of O(1) is

- A
  1
- B
  2
- C
  4
- D
  8

Correct Answer :A

Explanation

O(1) is known as constant time and O(1) is a flat line in terms of scalability. It will take the same amount of time

#108 Explained Report Bookmark

If n=4, then the value of O(n3) is

- A
  4
- B
  16
- C
  64
- D
  512

Correct Answer :C

Explanation

n=4

O(n^3) = 4^3 = 64

Note : if it is O(3n) then ans is 4 only instead of 12. Because asymptotic notations ignore constants.

#109 Not Explained Report Bookmark

If n=2, then the value of O(n) is

null

- A
  2
- B
  3
- C
  4
- D
  5

Correct Answer :A

No Explanation Available

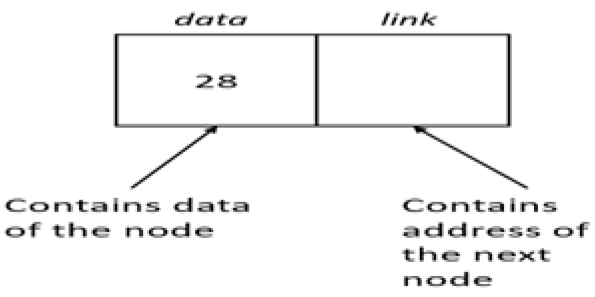
#110 Explained Report Bookmark

Each structure of a linked list consists _ _ _ _ _ _ _ no. of fields

- A
  2
- B
  3
- C
  4
- D
  1

Correct Answer :A

Explanation



A linked list consists of a data element known as a node. And each node consists of two fields: one field has data, and in the second field, the node has an address that keeps a reference to the next node.

#111 Explained Report Bookmark

Linked lists are not suitable for data structures of which one of the following problem?

- A
  insertion sort
- B
  Binary search
- C
  radix sort
- D
  polynomial manipulation problem

Correct Answer :B

Explanation

Binary search of the data structure link lists are not suitable.

Binary search based on divide and conquer algorithm, determination of middle element is important. Binary search is usually fast and efficient for arrays because accessing the middle index between two given indices is easy and fast. But memory allocation for singly linked list is dynamic and non-contiguous, which makes finding the middle element difficult. One approach could be of using skip list; one could be traversing the linked list using one pointer.

A linked list is a linear collection of data elements, whose order is not given by their physical placement in memory. Instead, each element points to the next. It is a data structure consisting of a collection of nodes which together represent a sequence.

#112Explained Report Bookmark

An item that is read as input can be either pushed to a stack and latter popped and printed or printed directly. Which of the following will be the output if the input is the sequence of items- 5, 6, 7, 8, 9?

- A
  7, 8, 9, 5, 6
- B
  7, 8, 9, 6, 5

- C
  9, 8, 7, 5, 6
- D
  5, 9, 6, 7, 8

Correct Answer :B

Explanation

The sequence given in option (C) is one of the only possible sequence which can be obtained.

We can obtain the sequence by performing operations in the manner:

Push 5

Push 6

Push 7

Pop 7

Push 8

Pop 8

Push 9

Pop 9

Pop 6

Pop 5.

hence the sequence will be 7,8,9,6,5.

#113 Explained Report Bookmark

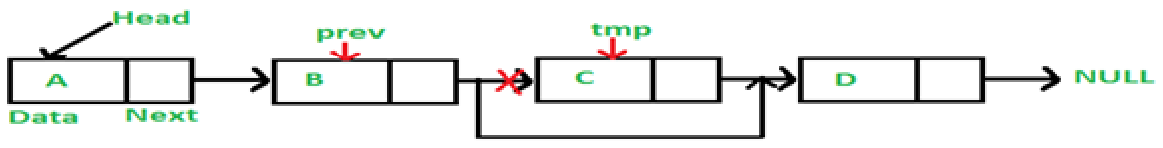
No.of pointers to be manipulated in a linked list to delete an item in the middle _ _ _

- A
  0
- B
  1
- C
  2
- D
  3

Correct Answer :C

Explanation

If we want to delete element which is at the end then we require 2 pointer variables of type node, 1st pointer variable is to locate last node (nodeA) and 2nd pointer is to locate the second last element (nodeB) so we can change the link of of nodeB to point to null



#114 Explained Report Bookmark

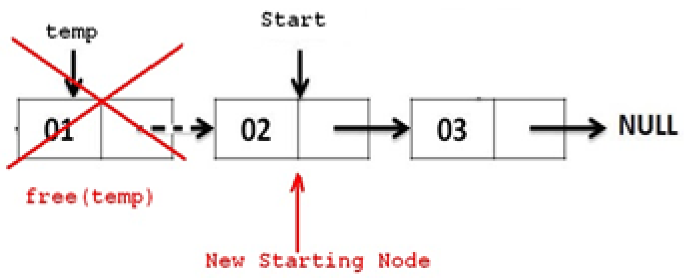
No.of pointers to be manipulated in a linked list to delete first item

- A
  0
- B
  1
- C
  2
- D
  3

Correct Answer :B

Explanation

In the Singly linked list if we want to delete elements from the beginning then we require only 1 pointer variable of type node to hold the first element.



#115 Explained Report Bookmark

Stack is useful for _ _ _ _ _ _ _

- A
  radix sort
- B
  readth first search
- C
  recursion
- D
  quick sort

Correct Answer :C

Explanation

Stack is used to store and restore the recursive function and its argument(s). To divide a problem into smaller pieces until reaching to solvable pieces. Then, to solve the problem by solving these small pieces of problem and merging the solutions to each other.

#116 Explained Report Bookmark

The end of the list is marked as

- A
  node.next=0
- B
  (node.last = 0)
- C
  node.next= &node;
- D
  node.previous=0;

Correct Answer :A

Explanation

Linked list can be visualized as a chain of nodes, where every node points to the next node. ... Last Link carries a Link as null to mark the end of the list.

#117 Explained Report Bookmark

LIFO is

- A
  stack

- B
  queue
- C
  linked list
- D
  tree

Correct Answer :A

Explanation

Why is a stack called LIFO? That means In stack which element entered at the last, which is deleted first. ... You take the last one on the stack. So, the last one on the stack is the first to go when you retrieve an item (i.e. LIFO)

#118 Explained Report Bookmark

A stack is most suitable to evaluate _ _ _ _ _ expression

- A
  postfix
- B
  prefix
- C
  infix
- D
  post & infix

Correct Answer :A

Explanation

An expression is called the postfix expression if the operator appears in the expression after the operands. It looks like this (operand1 operand2 operator).

To evaluate postfix expressions.

1) Create a stack to store operands or values.

2) Scan the given expression and do the following for every scanned element.

…..a) If the element is a number, push it into the stack

…..b) If the element is an operator, pop operands for the operator from stack. Evaluate the operator and push the result back to the stack

3) When the expression is ended, the number in the stack is the final answer

#119 Explained Report Bookmark

Which of the following data structure is suitable for priority queue?

- A
  Doubly linked list
- B
  Circular queues
- C
  Binary search
- D
  Heaps

Correct Answer :D

Explanation

Priority queue can be implemented using an array, a linked list, a heap data structure or a binary search tree. Among all these data structures, heap data structure provides an efficient implementation of priority queues.

Search Results

Priority queues are used in many algorithms like Huffman Codes, Prim's algorithm, etc. It is also used in scheduling processes for a computer, etc. Heaps are great for implementing a priority queue because of the largest and smallest element at the root of the tree for a max-heap and a min-heap

#120 Explained Report Bookmark

if front=rear ,then the queue is

- A
  full
- B
  empty
- C
  unknown value
- D
  1/2 full

Correct Answer :B

Explanation

rear shows underflow condition in a queue.

#121 Explained Report Bookmark

Suffix expression is

- A
  Infix
- B
  postfix
- C
  prefix
- D
  post & prefix

Correct Answer :B

Explanation

Whenever the operator is on right side of the operands then the expression is called suffix or postfix or reverse polish expression

#122 Explained Report Bookmark

To convert an Infix expression into postfix we require

- A
  stack
- B
  queue
- C
  linked list
- D
  dequeue

Correct Answer :A

Explanation

To convert infix expression to postfix expression, we will use the stack data structure. By scanning the infix expression from left to right, when we will get any operand, simply add them to the postfix form, and for the operator and parenthesis, add them in the stack maintaining the precedence of them.

#123 Explained Report Bookmark

To insert a node at the beginning of the single linked list _ _ _ _ _ _ _ no. of pointers to be manipulated

- A
  1

- B
  2
- C
  3
- D
  4

Correct Answer :A

Explanation

singly liked list has only a ‘next’ pointer, so only one pointer is to be manipulated.

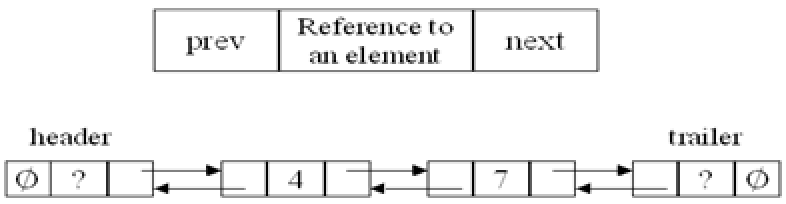
#124 Explained Report Bookmark

Doubly linked list uses _ _ _ _ _ _ _ _ no.of pointers

- A
  0
- B
  2
- C
  1
- D
  3

Correct Answer :B

Explanation



#125 Explained Report Bookmark

To insert a node at the beginning of the single linked list _ _ _ _ _ _ _ no. of pointers to be manipulated

- A
  1
- B
  0
- C
  2
- D
  3

Correct Answer :A

Explanation

singly liked list has only a ‘next’ pointer, so only one pointer is to be manipulated.

#126 Explained Report Bookmark

To insert a node at the end of the doubly linked list _ _ _ _ _ _ _ no. of pointers to be manipulated

- A
  1
- B
  2
- C
  3
- D
  4

Correct Answer :B

Explanation

A doubly linked list has two pointers ‘left’ and ‘right’ which enable it to traverse in either direction. Compared to singly liked list which has only a ‘next’ pointer, doubly linked list requires extra space to store this extra pointer. Every insertion and deletion requires manipulation of two pointers.

#127 Explained Report Bookmark

A tree is a _ _ _ _ _ data structure

- A
  non-recursive
- B
  recursive
- C
  linear
- D
  non-linear

Correct Answer :D

Explanation

A tree is a nonlinear data structure, compared to arrays, linked lists, stacks and queues which are linear data structures. A tree can be empty with no nodes or a tree is a structure consisting of one node called the root and zero or one or more subtrees.

#128 Explained Report Bookmark

A node that does not have any sub-tree is called a _ _ _ _ _ _ _

- A
  terminal node
- B
  root node
- C
  left node

- **D**
  right node

Correct Answer :A

Explanation

An item in any tree has two pointers or links: one to a left member, and another to a right ... *A node that does not have any sub-tree is called* as a terminal node

#129 Explained Report Bookmark

The number of edges in a regular graph of degree d and n vertices is

- **A**
  nd
- **B**
  n+d
- **C**
  maximum of n,d
- **D**
  nd/2

Correct Answer :D

Explanation

Consider the example of complete graph, In a complete graph which is $(n-1)(n-1)$ regular (where n is the no of vertices) has edges $n(n-1)2n(n-1)2$

nn vertices are adjacent to $n-1n-1$ vertices and an edge contributes two degree so dividing by 22.

d∗ n=2∗ |E|d∗ n=2∗ |E|

Therefore, in dd regular graph No of edges will be n∗ d/2

#130Explained Report Bookmark

A graph can be represented as an

- A
  Array
- B
  Structure
- C
  Union
- D
  Queue

Correct Answer :A

Explanation

Edge List Representation. Graphs can be represented using one dimensional array also. This is called the edge list. In this representation there are five edges are present, for each edge the first element is the source and the second one is the destination

#131Explained Report Bookmark

Which of he following is useful in traversing a given graph by breadth first search?

- A
  Stack
- B
  Set
- C
  List

- D
  Queue

Correct Answer :D

Explanation

Queue is used to implement traveral in Breadth First Search. Stack is used to implement traversal in Depth First Search.

#132 Explained Report Bookmark

Which of the following abstract data types can be used to represent a many to many relationships

- A
  tree
- B
  graph
- C
  queue
- D
  stack

Correct Answer :B

Explanation

Graph data structure is used to represent a many to many relationships.

#133 Explained Report Bookmark

To create a node dynamically in a singly linked list _ _ function in C is used

- A
  malloc

- B
  calloc
- C
  alloc
- D
  dealloc

Correct Answer :A

Explanation

malloc() is a function in C programming languageto create variables and objects dynamically.

#134Explained Report Bookmark

To delete an element from a queue we use the _ _ _ _ _ operation

- A
  pop
- B
  push
- C
  enqueue
- D
  dequeue

Correct Answer :D

Explanation

In Queue, accessing the content while removing it from the front end of the queue, is known as a Dequeue

#135Explained Report Bookmark

To add an element to a queue we use the _ _ _ _ _ operation

- A
  pop
- B
  push
- C
  enqueue
- D
  dequeue

Correct Answer :C

Explanation

The process to add an element into queue is called Enqueue and the process of removal of an element from queue is called Dequeue

#136 Explained Report Bookmark

A queue is ________ data structure.

- A
  FILO
- B
  LIFO
- C
  FIFO
- D
  LIFO

Correct Answer :C

Explanation

Because insertion and deletion takes place at different ends(rear end and front end). The element which is inserted first is removed first hence First in First out

#137 Explained Report Bookmark

A stack is __________ data structure.

null

- A
  Last in Last out
- B
  Last in First out
- C
  First in Last out
- D
  First in First out

Correct Answer :B

Explanation

Because insertion and deletion take place at the same end. The element which is pushed last is popped

first hence Last in First out

#138 Explained Report Bookmark

A linked list is _________ data structure

null

- A
  Static
- B
  Dynamic

- C
  Static or Dynamic
- D
  Global

Correct Answer :B

Explanation

Because memory is allocated to linked list at run-time(i.e when the program is active/running).

#139 Explained Report Bookmark

What is the worst-case time complexity of quick sort algorithm?

null

- A
  O(n/2)
- B
  O(2n
- C
  O(n)
- D
  none of above

Correct Answer :D

Explanation

Worst case time complexity of quick sort algorithm is O(n^2)

#140 Explained Report Bookmark

What is the average time complexity of the linear search algorithm?

- A
  O(n)
- B
  O(nlogn)
- C
  O(n/2)
- D
  O(n^2)

Correct Answer :A

Explanation

Worst case time complexity and average case time complexity of linear search algorithm are same

i.e. O(n). Best case time complexity of linear search algorithm is O(1).

#141 Explained Report Bookmark

Which sorting algorithm is efficient for array having small size

null

- A
  quick sort
- B
  insertion sort
- C
  bubble sort
- D
  merge sort

Correct Answer :B

Explanation

For sorting small arrays, insertion sort runs even faster than quick sort. But, it is impractical to sort large arrays.

#142 Not Explained Report Bookmark

Which of the following time complexities are generally same

null

- **A**
  best case & average case
- **B**
  worst case & average case
- **C**
  best case, average case & worst case
- **D**
  none of above

Correct Answer :B

No Explanation Available

#143 Explained Report Bookmark

The following postfix expression with single-digit operands is evaluating using a stack 8 2 3 ^ / 2 3 * + 5 1 * - Note that ^ is the exponential operator. The top two elements of the stack after the first * is evaluated are :

null

- **A**
  6,1
- **B**
  5,7
- **C**
  3,2
- **D**
  1,5

Correct Answer :A

Explanation

First 8 then 2 then 3 are pushed into the stacked.

On encountering ^ operator first 3 then 2 is popped from the stack and the operation 2^3 is performed.

Now 2^3 = 8 is pushed into the stack.

The stack contains data elements 8, 8

On encountering / operator. Two elements are popped from the stack hence the operation 8/8 is performed and (8/8 =1) the result 1 is pushed into the stack.

The stack now contains only one element i.e 1

Now data elements 2 and then 3 are pushed into the stack.

The Stack contains 1, 2, 3 data items.

On encountering the first * operator 2 and 3 are popped and the result 2*3 = 6 is pushed into the stack.

The stack contains 1,6 elements .

Topmost element is 6 while 1 is the second element from the top hence the answer is 6,1

#144 **Explained** **Report** **Bookmark**

The order of precedence (from highest to lowest) is ^,*,+,- The prefix expression corresponding to the infix expression a+b*c-d^e^f is

null

- A
  -+a*bc^^def
- B
  -+a*bc^de^f
- C
  ^+a*bc^-def
- D
  -+a^bc*^def

Correct Answer :A

Explanation

Here the game decider is not only the precedence of the operator but also the ASSOCIATIVITY of operator .

Given precedence order (from highest to lowest) is ^,*,+,-

There are two ^ operator so we need to consider their associativity.

Aassociativity of ^ operator is LEFT to RIGH.

It means d^e^f should be read as (d^e)^f.

Now we will convert infix expression to prefix expression by taking operands associated with the highest precedence operator

Step1.	a+b*c-d^e^f	->	a+b*c-(d^e)^f
	a+b*c-(d^e)^f	->	a+b*c-(^de)^f

Step2. a+b*c-(^de)^f -> a+b*c-(^^def)

Step3. a+b*c-(^^def) -> a+(b*c)-(^^def)

a+(b*c)-(^^def) -> a+(*bc) – (^^def)

step4. a+(*bc) – (^^def) -> (+a*bc) – (^^def)

(+a*bc) – (^^def) -> -+a*bc^^def

#145 Explained Report Bookmark

Which of the following tree traversal method gives the elements in a BST(Binary Search Tree) in descending order

- A
  postorder traversal
- B
  preorder traversal
- C
  inorder traversal
- D
  none of above

Correct Answer :D

Explanation

Using inorder traversal on a binary search tree will visit the values in ascending order.

#146 Explained Report Bookmark

Binary search Tree is balanced if

- A
  each node has exactly two child nodes
- B
  all leaf nodes are at same level
- C
  all nodes are balanced
- D
  root node is balanced

Correct Answer :C

Explanation

To check if a tree is height-balanced, get the height of left and right subtrees. Return true if difference between heights is not more than 1 and left and right subtrees are

balanced, otherwise return false.

#147 Explained Report Bookmark

What is the time complexity for searching an element in a binary search tree

- A
  O(n)
- B
  O(logn)
- C
  O(n^2)
- D
  O(2n)

Correct Answer :A

Explanation

Time complexity of binary search tree- Time complexity of BST operations is O(n) where n is the height of binary search tree.

#148 Explained Report Bookmark

Which of the following statement is false?

null

- **A**
  every tree is a bipartite graph
- **B**
  a tree contains a cycle
- **C**
  a tree with n nodes contains n-1 edges
- **D**
  a tree is a connected graph

Correct Answer :B

Explanation

A tree does not contains cycle while a graph may or may not contains a cycle.

A tree with cycle is called a Graph.

#149 Explained Report Bookmark

Given an undirected graph G with V vertices and E edges the sum of the degrees of all vertices is

null

- **A**
  E
- **B**
  2E

- C
  V
- D
  2V

Correct Answer :B

Explanation

You can solve this type of question by taking an example.

For example we have 2 vertices (2V) and 3 edges(3E) between these two vertices.

Degree of each vertex =3

Sum of degrees of both vertices = 6 which is twice the number of edges(3) hence we can conclude

Sum of the degrees of all the vertices is 2E.

#150 Explained Report Bookmark

Traversal of a graph is different from the tree because

- A
  there can be a loop in graph so we must maintain a visited flag for every vertex
- B
  DFS of a graph uses stack, but inorder traversal of a tree is recursive
- C
  of a graph uses queue, but a time efficient BFS of a tree is recursive
- D
  all the above

Correct Answer :A

Explanation

Traversal of a graph is different from tree because there can be a loop in the graph. We need to maintain an array to keep track of the vertices already visited.

#151 Explained Report Bookmark

Spanning tree of a graph contains minimum _____ no. of edges (where V= no. of vertices)

- A
  V-1
- B
  V+1
- C
  V
- D
  none of above

Correct Answer :A

Explanation

A spanning tree is a subset of graph G, which has all the vertices covered with minimum possible number of edges. To connect all the vertices of a graph, minimum number of

edges required will be V-1 where V is the number of vertices.

#152 Explained Report Bookmark

if there is an edge from any vertex to itself it is called as

- A
  self edge

- B
  cycle
- C
  loop
- D
  circuit

Correct Answer :C

Explanation

In a graph, if an edge is drawn from vertex to itself, it is called a loop.

#153Explained Report Bookmark

Separate chaining can be implemented by using

null

- A
  Linked List
- B
  Associative array
- C
  Both A&B
- D
  none of the above

Correct Answer :A

Explanation

In chaining each cell of hast table point to a linked list which can be implemented by using Linked List.

#154Explained Report Bookmark

Clustering problem may occur in

null

- A
  Separate Chaining
- B
  Open addressing
- C
  Linear Probing
- D
  Quadratic Probing

Correct Answer :C

Explanation

In Linear probing (probing means searching) many consecutive elements form groups and it starts taking time to find a free slot or to search an element. This grouping of elements is referred to Clustering.

#155 Explained Report Bookmark

Double Hashing is which one of the following type of collision handling method

- A
  Separate chaining
- B
  Linear probing
- C
  Open addressing
- D
  Quadratic Probing

Correct Answer :C

Explanation

There are two collision handling method

1. Chaining / Separate chaining
2. Open addressing:

Open addressing can be done using

1) Linear Probing

2) quadratic probing

3) Double Hashing

#156 Explained Report Bookmark

Which of the following technique is used to handle collision under Open Addressing

null

- A
  Linear Probing
- B
  Quadratic Probing
- C
  Double Hashing
- D
  All of the above

Correct Answer :D

Explanation

There are two collision handling method

1. Chaining / Separate chaining

2. Open addressing:

Open addressing can be done using

1) Linear Probing

2) quadratic probing

3) Double Hashing

#157 Explained Report Bookmark

An improvement on Direct Access Table is called as

null

- A
  Improved Direct Access Table
- B
  Upgraded Direct Access Table
- C
  Hash Table
- D
  None of the above

Correct Answer :C

Explanation

It can be improved by implementing hashing (hash function) on Direct Access Table. It is referred to as Hash table.

#158 Explained Report Bookmark

Which of the following is a dynamic implementation of hash table?

- A
  Separate chaining
- B
  open addressing
- C
  closed addressing
- D
  List chaining

Correct Answer :A

Explanation

Since Separate chaining is implemented by using Linked list which is dynamic in nature hence separate chaining is a dynamic implementation of hash table

#159 Explained Report Bookmark

Number of keys inserted into the hash table to the number of slots in a hash table is referred as

null

- A
  Insertion Ratio
- B
  Load factor
- C
  hashing factor
- D
  none of the above

Correct Answer :B

Explanation

Load Factor is defined as the ration of the numbers of keys inserted into the hash table to the number of slots in a hash table

#160 Explained Report Bookmark

Which of the following collision handling method is an efficient one

- A
  Open addressing
- B
  Separate chaining
- C
  Both A&B
- D
  none of the above

Correct Answer :C

Explanation

. open addressing is efficient storage-wise and separate chaining has efficient collision resolution.

#161 Explained Report Bookmark

Basic operation on hash table is

null

- A
  Insertion
- B
  Deletion
- C
  Probing
- D
  All of the above

Correct Answer :D

Explanation

Insertion , Deletion , Probing(Searching) can be performed on a hash table.

#162Explained Report Bookmark

Which of the following is an application of a hash table?

null

- A
  Cache Memory
- B
  File system
- C
  Database management system
- D
  none of the above

Correct Answer :A

Explanation

Cache memory is one of the major application of a hash table.

#163Explained Report Bookmark

For a given a hash table with25 slots that stores 2000 elements, then the load factor for a hash table is ________

null

- A
  80

- B
  0.0125
- C
  8000
- D
  1.25

Correct Answer :A

Explanation

Load Factor is defined as the ration of the numbers of keys inserted into the hash table to the number of slots in a hash table.

Load factor= 2000/25 = 80.

#164Explained Report Bookmark

Time taken for addition of element in queue is

null

- A
  O(1)
- B
  O(n)
- C
  O(log n)
- D
  none of the above

Correct Answer :A

Explanation

question is asked about the normal queue.

For priority queue, this should be O(log n)

#165 Explained Report Bookmark

We need to implement a queue using a circular array. If DATA is a circular array of CAPACITY elements, and rear is an index into that array, what will be the index for the element after rear?

null

- A
  (rear + 1) % CAPACITY
- B
  rear + (1 % CAPACITY)
- C
  rear % (1+CAPACITY)
- D
  (rear % 1) + CAPACITY

Correct Answer :A

Explanation

modulus of a will always be zero in case of integer value.

#166 Explained Report Bookmark

Which value is assigned/set at front and rear ends during the Initialization of a Queue?

- A
  0
- B
  1
- C
  -1

- D
  infinity

Correct Answer :C

Explanation

-1 is assigned at front and rear ends during the initialization of a queue because when queue is implemented using array then at that point of time array index will start at zero.

#167 Explained Report Bookmark

What should be the value of rear (end) if the queue is full (elements are completely occupied)?

- A
  1
- B
  - 1
- C
  MAX + 1
- D
  MAX - 1

Correct Answer :D

Explanation

When we insert the item in the queue, first we have to check that queue should not be full. For checking this condition we have to set that

if(rear==MAX-1)

{

```
        cout<< "queue is Full";

 }
```

Here MAX is the size of queue.

#168 Explained Report Bookmark

Which among the below specified condition is applicable if the Queue is non - empty?

- A
  rear > front
- B
  rear < front
- C
  rear = front
- D
  Unpredictable

Correct Answer :A

Explanation

The value of both front and rear will be -1 when the queue is empty. For insertion the value of rear is incremented by one and the element is inserted at new rear position. So if the queue is nonempty then rear > front condition must hold.

#169 Explained Report Bookmark

What is the ' next ' field of structure node in the Queue?

- A
  Results into the storage of queue elements.
- B
  Results into the storage of address of next node by holding the next element of queue.
- C
  Results into the memory allocation of data elements to next node.
- D
  Results into the address allocation data elements to next node.

Correct Answer :B

Explanation

the next filed store the address of next node which hold the next element of queue which help for moving forward

#170 Explained Report Bookmark

From where does the insertion and deletion of elements get accomplished in Queues?

- A
  Front & Rear ends respectively
- B
  Rear & Front ends respectively
- C
  Only Front ends
- D
  Only Rear ends

Correct Answer :B

Explanation

Insertion is performed from REAR end. Deletion is performed from FRONT end. Insertion operation is also known as ENQUEUE in queue.

#171 Explained Report Bookmark

Which of the following ' C ' functions is precise to verify the emptiness of a Queue?

A

```
int empty(Q*P)

{

  if (P->R==-1)

  return (1);

  return(0);
```

- }

B

```
int full (Q*P)

{

   if(P->R==MAX-1)

   return(1);

   return(0);
```

- }

C

```
int empty (Q*P)

{

   if (P<-R==-1)
```

```
    return(0);

    return(1);
```

- }

D

```
int full (Q*P)

{

    if(P<-R==MAX-1)

    return(0);

    return(1);
```

- }

Correct Answer :A

Explanation

In the given example P is the pointer object and rear is represented by variable R.

So if the queue is empty the value of rear should be -1.

int empty(Q *P)

{

if (P→R==-1)

```
return (1);

return(0);

}
```

Here P→R== -1 represents that the value of rear is -1 and queue is empty.

#172 Explained Report Bookmark

How are the elements with the same priority get processed according to the Priority Queue mechanism?

- A
  Before the processing of other elements with lower priority
- B
  After the processing of other elements with highest priority
- C
  On the basis of 'First-Come-First Served' priority
- D
  None of the Above

Correct Answer :C

Explanation

A priority queue is a collection of elements such that each element has been assigned a priority and such that the order in which elements are deleted and processed comes from the following rules: ... two elements with the same priority are processed according to the order in which they were added to the queue.(First-Come-First Served)

#173 Explained Report Bookmark

Which function plays an important role in returning the address of memory block allocated to locate / store a node especially while declaring ' top ' in the linked representation of the Stack?

- A
  galloc ()
- B
  falloc ()
- C
  malloc ()
- D
  calloc ()

Correct Answer :C

Explanation

malloc() stands for memory address and it is used a block of memory dynamically in c programming and it return void pointer

#174 Explained Report Bookmark

Which step is associated with an initialization process while converting a recursive into non - recursive algorithm?

- A
  Declaration of Stack
- B
  Pushing of all local variables and parameters into the stack
- C
  Assigning the values of formal parameters
- D
  Popping of return address

Correct Answer :A

Explanation

Rules for *converting*. a *recursive algorithm* to *non-recursive* one: Declare stack – It will hold local variables, parameters, return addresses, etc. The first statement after the stack *initialization* must-have. a label.

Click Here for more detail

#175 Explained Report Bookmark

Which data structures find their applications in BFS and DFS Traversal mechanisms on a Tree respectively?

- A
  Graph & Stack
- B
  Queue & Stack
- C
  Queue & Graph
- D
  None of the Above

Correct Answer :B

Explanation

BFS stands for Breadth first search and DFS stands for Depth first search. BFS uses queue data structure for finding the shortest path and DFS uses stack data structure.

DFS is an algorithm for finding or traversing graphs or trees in depth-ward direction. The execution of the algorithm begins at the root node and explores each branch before backtracking

BFS is an algorithm that is used to graph data or search trees or traversing structures. The algorithm efficiently visits and marks all the key nodes in a graph in breadthwise fashion.

#176 Explained Report Bookmark

How many elements are present in the stack if the variable top exhibits pointing towards the topmost element in the Array?

- A
  top +1
- B
  top - 1
- C
  zero
- D
  infinite

Correct Answer :A

Explanation

stack we start inserting the element from one end called top and the 1st element stored at 0 index and when top =-1 that shows the empty stack and while inserting the elements top get incremented , so if the top exhibit pointing toward the topmost element in the array it means we have top+1 element

0 1 2 3 4

10 20 30 40 50

Top

It means we have 5 elements in an array

#177 Explained Report Bookmark

Which expressions are also regarded as ' Reverse Polish Notations '?

- A
  Prefix
- B
  Postfix
- C
  Infix
- D
  All of the above

Correct Answer :B

Explanation

In data structure reverse polish notation is also called as polish postfix notation in which operator follow the operands and in polish notation operators come before the operands

Example of reverse polish notation : AB+

#178 Explained Report Bookmark

When is the pop operation on Stack considered to be an error?

- A
  Only when the stack is empty
- B
  Only when the stack is full
- C
  When the stack is neither empty nor full
- D
  Cannot be predicted

Correct Answer :A

Explanation

when the stack is empty and we try to pop operation on an empty stack it will give an error message . pop means to delete the element from the stack but we have empty stack , no element available in the stack so it will give error

#179 Explained Report Bookmark

What does the push operation top = top +1 indicate while representing the stack in one - dimensional array?

- A
  Stack Growing
- B
  Stack Shrinking
- C
  Stack Stability
- D
  Stack Instability

Correct Answer :A

Explanation

The stack is a linear data structure that follows the LIFO (last in first out) algorithm. And the first element store at 0th index and while pushing element top get incremented by 1 position (top= top+ 1) and it means stack growing

#180 Explained Report Bookmark

Which of the following techniques represents the precise sequence of an In - Order Traversal of a Binary Tree?

- A
  Visit the Root, Traverse Left Subtree, Traverse Right Subtree
- B
  Traverse Left Subtree, Visit the Root, Traverse Right Subtree

- C
  Traverse Left Subtree, Traverse Right Subtree, Visit the Root
- D
  None of the Above

Correct Answer :B

Explanation

Traversing a binary tree means visiting each node of a tree exactly once. Traversal of tree gives the linear order of nodes. There are three types of traversal in a binary tree. Preorder, Inorder and Postorder traversal.

Algorithm for inorder traversal.

- Traverse left subtree of root in inorder.

- Visit the Root.

- Traverse right subtree of root in inorder.

#181 Explained Report Bookmark

What does a node possessing zero degree in Trees known as?

- A
  Branch Node
- B
  Root Node
- C
  Leaf Node
- D
  Trunk Node

Correct Answer :C

Explanation

In other words, the degree is the number of descendants of a node. If the degree is zero, it is called a terminal or leaf node of a tree.

#182 Explained Report Bookmark

Which type of linked list comprises the adjacently placed first and the last elements?

- A
  Singly Linked List
- B
  Doubly Linked List
- C
  Circular Linked List
- D
  All of the above

Correct Answer :C

Explanation

In a circular linked list the first element points to the last node and the last element point to the first node to form a circle and it is placed adjacently to each other. In this linked list there is no null node at the end

#183 Explained Report Bookmark

Traversal of a linked list always starts from the __________.

- A
  First Node
- B
  Middle Node

- C
  Last Node
- D
  None of the Above

Correct Answer :A

Explanation

Traversal of a linked list always starts from the First Node.

Example:

struct node

{

Int data;

struct node ∗ link;

} ∗ start;

For traversing a single link list we will take a pointer P of type struct node and initialized with start.

P=start;

Now P points to the first node of link list. Link list can be traversed as follows.

while(P ! NULL)

```
{
cout << (P→ data);
P = P→link;
}
```

#184 Explained Report Bookmark

Which type of linked list / s comprise / s a node containing a pointer to predecessor as well as successor?

- A
  linear Linked List
- B
  Circular Linked List
- C
  Doubly Linked List
- D
  None of the above

Correct Answer :C

Explanation

Doubly link list contains pointers to predecessor as well as successor. In Doubly link list each node has two pointers. One pointer points to the next node and the other points to the previous node.

Example:

struct node

```
{
    Int data;
    struct node *prev;
    struct node *next;
}
```

In the above example, pointer named prev will contain the address of previous node and next pointer will contain the address of next node. By using these two pointers we can move in both directions at an

#185 Explained Report Bookmark

Doubly Linked List is especially applicable, for which of the following implementation?

- A
  Double Ended Queue
- B
  Linear List of Integers
- C
  Variable Length String Data Structure
- D
  All of the Above

Correct Answer :D

Explanation

In doubly linked list we can travel in both direction forward as well as backward Doubly linked list is a collection of nodes linked together in a linear order . Each node of the list contains two parts data part

Address of previous node	Data	Address of next node

Doubly linked list is also used to implement stack , heap and binary tree

#186 Explained Report Bookmark

How many pointers are necessarily changed for the insertion in a Linked List?

- A
  One
- B
  Two
- C
  Three
- D
  Five

Correct Answer :B

Explanation

This depends on whether we are inserting the new node in the middle of the list (surrounded by two nodes), or at the head or tail of the list. Insertion at the middle node will affect 4 pointers whereas at the head or tail will affect only 2 pointers.

#187 Explained Report Bookmark

Which of the following is the Worst-case of Quick Sort? -

- A
  O (n log n)
- B
  O (N^2)
- C
  O (log n)
- D
  O (n^2 / 4)

Correct Answer :B

Explanation

The worst case time complexity of a typical implementation of QuickSort is O(n^2). The worst case occurs when the picked pivot is always an extreme (smallest or largest) element. This happens when input array is sorted or reverse sorted and either first or last element is picked as pivot.

#188 Explained Report Bookmark

Which of the following data structure is not a Linear Data Structure?

- A
  Arrays
- B
  Linked lists
- C
  Both of these
- D
  None of these

Correct Answer :D

Explanation

Array and linked list both are linear data structures which means the data is organized in linear order and all elements are connected to one after the other. Array , stack , queue and linked list all are linear data structure

#189 Explained Report Bookmark

The operation of processing each element in the list is known as, _________.

- A
  Sorting
- B
  Merging
- C
  Inserting
- D
  Traversal

Correct Answer :D

Explanation

The operation of processing each element in the list is known as. The operation of processing each element in the list is known as traversal. It refers to the process of visiting (checking and/or updating) each node in a tree data structure, exactly once

#190 Explained Report Bookmark

You have to sort a list L consisting of a sorted list, followed by a few “ random ” elements. Which of the following sorting methods would be especially suitable for such a task?

- A
  Bubble sort
- B
  Selection sort
- C
  Quick sort

- D
  Insertion sort

Correct Answer :D

Explanation

Insertion sort is a sorting algorithm in which the elements are transferred one at a time to the right position and this sorting algorithm is not efficient for larger data set

#191 Explained Report Bookmark

Which Linked List have the last node of the list pointing to the first node?

- A
  Circular Doubly Linked List
- B
  Circular Singly Linked List
- C
  Circular Linked List
- D
  Doubly Linked List

Correct Answer :C

Explanation

circular linked list is a linked list in which last node always point to the first node i.e. last node contains the address of the first node.

#192 Explained Report Bookmark

Which if the following is/are the levels of implementation of data structure

- A
  Abstract level
- B
  Application level
- C
  Implementation level
- D
  All of the above

Correct Answer :D

Explanation

All are the levels of implementation of data structure.

#193 Explained Report Bookmark

A binary search tree whose left subtree and right subtree differ in height by at most 1 unit is called

- A
  AVL tree
- B
  Red-black tree
- C
  Lemma tree
- D
  None of the above

Correct Answer :A

Explanation

An AVL tree is the binary search tree in which every sub tree is an AVL tree but is not entirely balanced. It takes less time to calculate the height of sub trees in the AVL tree which is the main advantage of these trees. So it is widely used in computer algorithms in a design method. The various operations on AVL trees such as insertion, deletions, searching, etc.

#194 Explained Report Bookmark

…………… is not the component of data structure

- A
  Operations
- B
  Storage Structures
- C
  Algorithms
- D
  None of above

Correct Answer :D

Explanation

All 3 above is the component of data structure . In data structure operation like insert , delete etc and storage structure are the representation of particular data structure in main memory

Algorithm in data structure are the step by step procedure to solve a problem and get the desired output

#195 Explained Report Bookmark

Inserting an item into the stack when stack is not full is called …………. Operation and deletion of item form the stack, when stack is not empty is called ………..operation.

- A
  push, pop

- B
  pop, push
- C
  insert, delete
- D
  delete, insert

Correct Answer :A

Explanation

A stack is a simple last-in, first-out (LIFO) data structure. That is, the last data element stored in a stack is the first data element retrieved from the stack. The common analogy is a stack of plates in a cafeteria: when you go through the line, you pop the top plate off the stack; the dish washer (stepping away from reality a bit), pushes a single clean plate on top of the stack.

push() function is used to insert an element at the top of the stack.

The element is added to the stack container and the size of the stack is increased by 1.

pop() function is used to remove an element from the top of the stack(newest element in the stack).

The element is removed to the stack container and the size of the stack is decreased by 1.

#196 Explained Report Bookmark

………… is very useful in situation when data have to stored and then retrieved in reverse order.

- A
  Stack

- B
  Queue
- C
  List
- D
  Link list

Correct Answer :A

Explanation

In stack all operations are performed on one end which is called TOP.and when we retrieve the data from stack the we get in reverse order.

and suppose we have pushed some elements in the stack in sequence - 1 4 6 7

7

6

4

1

And when we pop , it will delete from the top of the stack and we get reverse order 7 6 4 1

#197 Explained Report Bookmark

Which data structure allows deleting data elements from and inserting at rear?

- A
  Stacks
- B
  Queues
- C
  Dequeues
- D
  Binary search tree

Correct Answer :B

Explanation

Queue is a linear data structure in which we can perform operations from both ends. Generally we insert a new element from the end which is called rear and delete the element from beginning which is called front.

Front
rear

#198 Explained Report Bookmark

Which of the following is non-liner data structure

- A
  Stacks

- B
  List
- C
  Strings
- D
  Trees

Correct Answer :D

Explanation

In the non-linear data structure the data elements are not stored in a sequential manner rather the elements are hierarchically related. ... Examples of the linear data structure are array, queue, stack, linked list, etc. In contrast, tree and graph are the examples of the non-linear data structure

#199 Explained Report Bookmark

To represent hierarchical relationship between elements, Which data structure is suitable

- A
  Dequeue
- B
  Priority
- C
  Tree
- D
  Graph

Correct Answer :C

Explanation

A tree is a non-linear data structure which is best suitable for representing hierarchical relationship between elements.

#200 Explained Report Bookmark

A graph is a collection of nodes, called ………. And line segments called arcs or ……….. that connect pair of nodes.

- A
  vertices, edges
- B
  edges, vertices
- C
  vertices, paths
- D
  graph node, edges

Correct Answer :A

Explanation

Nodes in a graph are known as vertices and connecting lines of those nodes are known as edges.

#201 Explained Report Bookmark

The time complexity of linear search algorithm over an array of n elements is

null

- A
  O $(\log_2 n)$
- B
  O(n)
- C
  O$(n \log_2 n)$
- D
  0 (n_2)

Correct Answer :B

Explanation

Linear search has linear-time complexity; binary search has log-time complexity.

Here is a table that provides some intuition about the running speeds of algorithms

Logarithmic: Linear:

array size	N
8	8
128	128
256	256
1000	1000
100,000	100,000 ....

Binary search and other divide-and-conquer algorithms have logN time complexity; we say O(logN)

Linear search have linear time complexity: O(N). So the option is 'B'

#202 Explained Report Bookmark

To sort many large object or structures, it would be most efficient to

null

- **A**
  Place reference to them in and array an sort the array
- **B**
  Place them in a linked list and sort the linked list
- **C**
  Place pointers to them in an array and sort the array
- **D**
  Place them in an array and sort the array

Correct Answer :B

Explanation

- Dynamic structure (Memory Allocated at run-time).
- We can have more than one datatype.
- Re-arrange of linked list is easy (Insertion-Deletion).
- It doesn't waste memory.

So, the answer is 'B'

#203 Explained Report Bookmark

Average successful search time for sequential search on 'n' items is

null

- **A**
  n/2
- **B**
  (n-1)/2
- **C**
  (n+1)/2
- **D**
  None of these

Correct Answer :C

Explanation

If search key matches the very first item, with one comparison we can terminate.

If it is second, two comparisons, etc.

Average =[n (n + 1)] / 2

#204 Explained Report Bookmark

Which of the following sorting procedure is the slowest ?

- A
  Quick sort
- B
  Heap sort
- C
  Shell sort
- D
  Bubble sort

Correct Answer :D

Explanation

Bubble sort’s sorting procedure is the slowest one among rest of the given procedures.

#205 Explained Report Bookmark

Which of the following best describes sorting ?

- A
  Accessing and processing each record exactly once
- B
  Finding the location of the record with a given key

- C
  Arranging the data (record) in some given order
- D
  Adding a new record to the data structure

Correct Answer :C

Explanation

Sorting refers to ordering data in an increasing or decreasing fashion according to some linear relationship among the data items.

#206 Explained Report Bookmark

A sort which compares adjacent elements in a list and switches where necessary is

- A
  insertion sort
- B
  heap sort
- C
  quick sort
- D
  bubble sort

Correct Answer :D

Explanation

A sort which compares adjacent elements in a list and switches where necessary is a bubble sort. Bubble sort is a simple sorting algorithm that repeatedly steps through the list to be sorted, compares each pair of adjacent items and swaps them if they are in the wrong order.

#207 Explained Report Bookmark

. ………………. is not an operation performed on linear list

- A
  Insertion
- B
  Insertion & Deletion
- C
  Deletion & Traversal
- D
  None of the above

Correct Answer :D

Explanation

All the operations are performed on linear list.

#208 Explained Report Bookmark

Which is/are the application(s) of stack

- A
  Function calls
- B
  Large number Arithmetic
- C
  Evaluation of arithmetic expressions
- D
  All

Correct Answer :D

Explanation

A stack is an abstract data types that serves as a collection of elements, with two main principal operation :1) push, which adds an element to the collection, and 2) pop, which removes the most recently added elements that was not yet removed.

#209Explained Report Bookmark

What is the advantage of bubble sort over other sorting techniques?

null

- A
  It is faster
- B
  Consumes less memory
- C
  Detects whether the input is already sorted
- D
  All of the mentioned

Correct Answer :C

Explanation

Bubble sort is one of the simplest sorting techniques and perhaps the only advantage it has over other techniques is that it can detect whether the input is already sorted.

#210 Explained Report Bookmark

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

Correct Answer :A

Explanation

Concept

Bubble sort repeatedly steps through the list to be sorted, compares each pair of adjacent items and swaps them if they are in the wrong order. The pass through the list is repeated until no swaps are needed, which indicates that the list is sorted.

Explanation

Array elements: 1,2,4,3

1st pass = 1,2,4,3

1 and 2 not swap:

2nd pass = 1,2,4,3

2 and 4 not swap:

3rd pass = 1,2,4,3

4 and 3 swap:

Since array is sorted after 3th pass

Hence the answer is Option 4

#211 Explained Report Bookmark

What is the best case complexity of QuickSort?

null

- A
  O(nlogn)
- B
  O(logn)
- C
  O(n)
- D
  O(n2)

Correct Answer :A

Explanation

The array is partitioned into equal halves, using the Divide and Conquer master theorem, the complexity is found to be O(nlogn).

#212 Explained Report Bookmark

The given array is arr = {2,3,4,1,6}. What are the pivots that are returned as a result of subsequent partitioning?

null

- A
  1 and 3
- B
  3 and 1
- C
  2 and 6
- D
  6 and 2

Correct Answer :A

Explanation

The call to partition returns 1 and 3 as the pivot elements.

#213 Explained Report Bookmark

What is the average case complexity of selection sort?

- A
  O(nlogn)
- B
  O(logn)
- C
  O(n)
- D
  O(n2)

Correct Answer :D

Explanation

In the average case, even if the input is partially sorted, selection sort behaves as if the entire array is not sorted. Selection sort is insensitive to input.

#214 Explained Report Bookmark

What is the disadvantage of selection sort?

null

- A
  It requires auxiliary memory
- B
  It is not scalable
- C
  It can be used for small keys
- D
  None of the mentioned

Correct Answer :B

Explanation

As the input size increases, the performance of selection sort decreases.

#215 Explained Report Bookmark

What is the worst case complexity of bubble sort?

null

- A
  O(nlogn)

- **B**
  O(logn)
- **C**
  O(n)
- **D**
  O(n2)

Correct Answer :D

Explanation

Bubble sort works by starting from the first element and swapping the elements if required in each iteration.

#216 Explained Report Bookmark

What is the worst case complexity of selection sort?

null

- **A**
  O(nlogn)
- **B**
  O(logn)
- **C**
  O(n)
- **D**
  O(n2)

Correct Answer :D

Explanation

Selection sort creates a sub-list, LHS of the ‘min’ element is already sorted and RHS is yet to be sorted. Starting with the first element the ‘min’ element moves towards the final element.

#217Explained Report Bookmark

Process of inserting an element in stack is called ____________

null

- A
  Create
- B
  Push
- C
  Evaluation
- D
  Pop

Correct Answer :B

Explanation

Push operation allows users to insert elements in stack. If stack is filled completely and trying to perform push operation stack – overflow can happen.

#218Explained Report Bookmark

Process of removing an element from stack is called __________

null

- A
  insert
- B
  Push
- C
  delete
- D
  Pop

Correct Answer :D

Explanation

Elements in stack are removed using pop operation. Pop operation removes the top most element in the stack i.e. last entered element.

#219 Explained Report Bookmark

In a stack, if a user tries to remove an element from empty stack it is called _________

null

- A
  Underflow
- B
  Empty collection
- C
  Overflow
- D
  Garbage Collection

Correct Answer :A

Explanation

Underflow occurs when the user performs a pop operation on an empty stack. Overflow occurs when the stack is full and the user performs a push operation. Garbage Collection is used to recover the memory occupied by objects that are no longer used.

#220 Explained Report Bookmark

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

null

- A
  Overflow
- B
  Crash
- C
  Underflow
- D
  User flow

Correct Answer :A

Explanation

The stack is filled with 5 elements and pushing one more element causes a stack overflow. This results in overwriting memory, code and loss of unsaved work on the computer.

#221 Explained Report Bookmark

Entries in a stack are “ordered”. What is the meaning of this statement

null

- A
  A collection of stacks is sortable
- B
  Stack entries may be compared with the ‘<‘ operation
- C
  The entries are stored in a linked list
- D
  There is a Sequential entry that is one by one

Correct Answer :D

Explanation

In stack data structure, elements are added one by one using push operation. Stack follows LIFO Principle i.e. Last In First Out(LIFO).

#222 Explained Report Bookmark

Which of the following applications may not use a 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

Correct Answer :D

Explanation

Data transfer between the two asynchronous process uses the queue data structure for synchronisation. The rest are all stack applications.

#223 Explained Report Bookmark

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?

null

- A
  1
- B
  2
- C
  3

- D
  4

Correct Answer :B

Explanation

In the entire parenthesis balancing method when the incoming token is a left parenthesis it is pushed into stack. A right parenthesis makes pop operation to delete the elements in stack till we get left parenthesis as top most element. 2 left parenthesis are pushed whereas one right parenthesis removes one of left parenthesis. 2 elements are there before right parenthesis which is the maximum number of elements in stack at run time.

#224 Explained Report Bookmark

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?

null

- A
  1
- B
  2
- C
  3
- D
  4

Correct Answer :D

Explanation

When we perform the conversion from infix to postfix expression +, *, (, * symbols are placed inside the stack. A maximum of 4 symbols are identified during the entire conversion.

#225 Explained Report Bookmark

The data structure required to check whether an expression contains balanced parenthesis is?

null

- A
  Stack
- B
  Queue
- C
  Array
- D
  Tree

Correct Answer :A

Explanation

The stack is a simple data structure in which elements are added and removed based on the LIFO principle. Open parenthesis is pushed into the stack and a closed parenthesis pops out elements till the top element of the stack is its corresponding open parenthesis. If the stack is empty, parenthesis is balanced otherwise it is unbalanced.

#226 Explained Report Bookmark

The process of accessing data stored in a serial access memory is similar to manipulating data on a ________

null

- A
  Heap
- B
  Binary Tree
- C
  Array
- D
  Stack

Correct Answer :D

Explanation

In serial access memory data records are stored one after the other in which they are created and are accessed sequentially. In stack data structure, elements are accessed sequentially. Stack data structure resembles the serial access memory.

#227 Explained Report Bookmark

Which data structure is needed to convert infix notation to postfix notation?

null

- A
  Branch
- B
  Tree
- C
  Queue
- D
  Stack

Correct Answer :D

Explanation

The Stack data structure is used to convert infix expression to postfix expression. The purpose of stack is to reverse the order of the operators in the expression. It also serves as a storage structure, as no operator can be printed until both of its operands have appeared.

#228 Explained Report Bookmark

Which data structure is used for implementing recursion?

null

- A
  Queue
- B
  Stack
- C
  Array
- D
  List

Correct Answer :B

Explanation

stacks are used for the implementation of Recursion.

#229 Explained Report Bookmark

Consider the following operation performed on a stack of size 5.

Push(1);

Pop();

Push(2);

Push(3);

Pop();

Push(4);

Pop();

Pop();

Push(5);

After the completion of all operation, the number of elements present in stack are

- A
  1
- B
  2
- C
  3
- D
  4

Correct Answer :A

Explanation

Number of elements present in stack is equal to the difference between number of push operations and number of pop operations. Number of elements is 5-4=1.

#230 Explained Report Bookmark

Which of the following is not an inherent application of stack?

- A
  Reversing a string
- B
  Evaluation of postfix expression
- C
  Implementation of recursion

- D
  Job scheduling

Correct Answer :D

Explanation

Job Scheduling is not performed using stacks.

#231 Explained Report Bookmark

If the elements “A”, “B”, “C” and “D” are placed in a stack and are deleted one at a time, what is the order of removal?

null

- A
  ABCD
- B
  DCBA
- C
  DCAB
- D
  ABDC

Correct Answer :B

Explanation

Stack follows LIFO(Last In First Out). So the removal order of elements are DCBA.

#232 Explained Report Bookmark

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 ?

null

- A
  Queue
- B
  Stack
- C
  Tree
- D
  Linked list

Correct Answer :A

Explanation

Linear list of elements in which deletion is done at front side and insertion at rear side is called Queue. In stack we will delete the last entered element first.

#233 Explained Report Bookmark

The data structure required for Breadth First Traversal on a graph is?

null

- A
  Stack
- B
  Array
- C
  Queue
- D
  Tree

Correct Answer :C

Explanation

In Breadth First Search Traversal, BFS, starting vertex is first taken and adjacent vertices which are unvisited are also taken. Again, the first vertex which was added as an unvisited adjacent vertex list will be considered to add further unvisited vertices of the graph. To get first unvisited vertex we need to follows First In First Out principle. Queue uses FIFO principle.

#234 Explained Report Bookmark

A data structure in which elements can be inserted or deleted at/from both the ends but not in the middle is?

null

- A
  Queue
- B
  Circular queue
- C
  Dequeue
- D
  Priority queue

Correct Answer :C

Explanation

In dequeuer, we can insert or delete elements from both the ends. In queue, we will follow first in first out principle for insertion and deletion of elements. Element with least priority will be deleted in a priority queue.

#235 Explained Report Bookmark

A normal queue, if implemented using an array of size MAX_SIZE, gets full when

null

- A
  Rear = MAX_SIZE – 1

- B
  Front = (rear + 1)mod MAX_SIZE
- C
  Front = rear + 1
- D
  Rear = front

Correct Answer :A

Explanation

When Rear = MAX_SIZE – 1, there will be no space left for the elements to be added in queue. Thus queue becomes full.

#236 Explained Report Bookmark

Which of the following is not the type of queue?

null

- A
  Ordinary queue
- B
  Single ended queue
- C
  Circular queue
- D
  Priority queue

Correct Answer :B

Explanation

Queue always has two ends. So, single ended queue is not the type of queue.

#237 Explained Report Bookmark

In a circular queue, how do you increment the rear end of the queue?

null

- A
  rear++
- B
  (rear+1) % CAPACITY
- C
  (rear % CAPACITY)+1
- D
   rear–

Correct Answer :B

Explanation

 Ensures rear takes the values from 0 to (CAPACITY-1).

#238 Explained Report Bookmark

What is the term for inserting into a full queue known as?

- A
  overflow
- B
  underflow
- C
  null pointer exception
- D
  program won’t be compiled

Correct Answer :A

Explanation

Overflow happens when we try to push more items than it can hold. And here we have full queue and if we try to insert new element in the queue we get “Overflow” message

#239 Explained Report Bookmark

What is the time complexity of enqueue operation?

- A
  O(logn)
- B
  O(nlogn)
- C
  O(n)
- D
  O(1)

Correct Answer :D

Explanation

Enqueue operation used in queue data structure which means to insert the new element in queue , and in queue we have two end from one end it insert the element (enqueue) called rear and from second end it delete the element (dequeue) called front

And the time complexity to enqueue and dequeue the element is O(1).

#240 Explained Report Bookmark

What does the following piece of code do?

```
public Object function()

{
     if(isEmpty())

     return -999;

     else

     {

             Object high;

             high = q[front];

             return high;

     }

}
```

- **A**
  Dequeue
- **B**
  Enqueue
- **C**
  Return the front element
- **D**
  Return the last element

Correct Answer :C

Explanation

q[front] gives the element at the front of the queue, since we are not moving the ‘front’ to the next element,

it is not a dequeue operation.

#241 Explained Report Bookmark

What is the need for a circular queue?

- A
  effective usage of memory
- B
  easier computations
- C
  to delete elements based on priority
- D
  implement LIFO principle in queues

Correct Answer :C

Explanation

In a linear queue, dequeue operation causes the starting elements of the array to be empty, and there is no way you can use that space, while in a circular queue, you can effectively use that space. Priority queue is used to delete the elements based on their priority. Higher priority elements will be deleted first whereas lower priority elements will be deleted next. Queue data structure always follows FIFO principle.

#242 Explained Report Bookmark

In linked list implementation of queue, if only front pointer is maintained, which of the following operation take worst case linear time?

null

- A
  Insertion
- B
  Deletion
- C
   To empty a queue

- D
  Both Insertion and To empty a queue

Correct Answer :D

Explanation

Since front pointer is used for deletion, so worst time for the other two cases.

#243 Explained Report Bookmark

In linked list implementation of a queue, where does a new element be inserted?

null

- A
  At the head of link list
- B
  At the centre position in the link list
- C
  At the tail of the link list
- D
  At any position in the linked list

Correct Answer :C

Explanation

Since queue follows FIFO so new element inserted at last.

#244 Explained Report Bookmark

In linked list implementation of a queue, front and rear pointers are tracked. Which of these pointers will change during an insertion into a NONEMPTY queue?

- A
  Only front pointer
- B
   Only rear pointer
- C
   Both front and rear pointer
- D
  No pointer will be changed

Correct Answer :B

Explanation

Since queue follows FIFO so new element inserted at last.

 In a queue , the rear is generally used for inserting the element and suppose if we decide to insert it from one end (depend on user) , then we delete it from the other end . but during insertion only rear pointer will change

#245 Explained Report Bookmark

. To implement a stack using queue(with only enqueue and dequeue operations), how many queues will you need?

null

- A
  1
- B
  2
- C
  3
- D
  4

Correct Answer :B

Explanation

Either the push or the pop has to be a costly operation, and the costlier operation requires two queues.

#246Explained Report Bookmark

If we choose Prim's Algorithm for uniquely weighted spanning tree instead of Kruskal's Algorithm, then

null

- A
  we'll get a different spanning tree.
- B
  we'll get the same spanning tree.
- C
  spanning will have less edges.
- D
  spanning will not cover all vertices.

Correct Answer :B

Explanation

Regardless of which algorithm is used, in a graph with unique weight, resulting spanning tree will be same.

#247Explained Report Bookmark

Which of the following statement is false

- A
  Arrays are dense lists and static data structure
- B
  data elements in a linked list need not be stored in adjacent space in memory

- C
  pointers store the next data element of a list
- D
  linked lists are a collection of the nodes that contain information part and next pointer

Correct Answer :C

Explanation

The Pointer in C, is a variable that stores address of another variable. A pointer can also be used to refer to another pointer function. A pointer can be incremented/decremented, i.e., to point to the next/ previous memory location. The purpose of pointer is to save memory space and achieve faster execution time

#248 Explained Report Bookmark

Which of the following data structures are indexed structures?

- A
  Linear arrays
- B
  Linked lists
- C
  Queue
- D
  Stack

Correct Answer :A

Explanation

Array data structures have indexes to access elements randomly.

#249 Explained Report Bookmark

When new data are to be inserted into a data structure, but there is no available space; this situation is usually called

- A
  underflow
- B
  overflow
- C
  Housefull
- D
  saturated

Correct Answer :B

Explanation

It means that when there is no blank space to store any value, Then this situation is called an "overflow situation

#250 Explained Report Bookmark

The situation when in a linked list START=NULL is

- A
  underflow
- B
  overflow
- C
  Housefull
- D
  saturated

Correct Answer :A

Explanation

if the start pointer contains null inside it then it means that there is no node in the linked list to which the pointer should point and therefore it contains null to indicate that the list is empty or the underflow condition.

#251 Explained Report Bookmark

Which of the following data structure can’t store the non-homogeneous data elements?

- A
  Arrays
- B
  Records
- C
  Pointers
- D
  None

Correct Answer :A

Explanation

Arrays is the data structure which cannot store the non-homogeneous data elements.

#252 Explained Report Bookmark

Which of the following data structures store the homogeneous data elements?

- A
  Arrays
- B
  Records
- C
  Pointers
- D
  None

Correct Answer :A

Explanation

An array is a data structure that contains a group of elements. Typically these elements are all of the same data type or homogeneous, such as an integer or string. Arrays are commonly used in computer programs to organize data so that a related set of values can be easily sorted or searched.

#253 Explained Report Bookmark

Which of the following is not a limitation of the binary search algorithm?

- A
  must use a sorted array
- B
  the requirement of a sorted array is expensive when a lot of insertion and deletions are needed
- C
  there must be a mechanism to access the middle element directly
- D
  a binary search algorithm is not efficient when the data elements are more than 1000

Correct Answer :D

Explanation

This is because there is no restriction on the number of elements when being searched using binary search algorithm.

#254 Explained Report Bookmark

The binary search algorithm can not be applied to

- A
  sorted linked list
- B
  sorted binary trees
- C
  sorted linear array
- D
  pointer array

Correct Answer :A

Explanation

The binary search algorithm cannot be applied to sorted linked list

#255 Explained Report Bookmark

When determining the efficiency o the algorithm, the space factor is measured by

- A
  Counting the maximum memory needed by the algorithm
- B
  Counting the minimum memory needed by the algorithm
- C
  Counting the average memory needed by the algorithm
- D
  Counting the maximum disk space needed by the algorithm

Correct Answer :A

Explanation

The correct answer is A, Counting the maximum memory needed by the algorithm. When X is an algorithm, space is one of the factors used by X to determine how efficient X is. In a space factor, space can be measured. To measure it, one needs to count how much memory space the algorithm needs.

#256 Explained Report Bookmark

The complexity of Bubble sort algorithm is

- A
  O(n)
- B
  O(n2)
- C
  O(log n)
- D
  O(n log n)

Correct Answer :B

Explanation

The following table summarizes the time complexities of bubble sort in each case-

Worst Case-

- In worst case, the outer loop runs O(n) times.
- Hence, the worst case time complexity of bubble sort is O(n x n) = O(n2).

Best Case-

- In best case, the array is already sorted but still to check, bubble sort performs O(n) comparisons.
- Hence, the best case time complexity of bubble sort is O(n).

Average Case-

- In average case, bubble sort may require (n/2) passes and O(n) comparisons for each pass.

- Hence, the average case time complexity of bubble sort is O(n/2 x n) = $\Theta(n^2)$.

	Time Complexity
Best Case	O(n)
Average Case	$\Theta(n^2)$
Worst Case	$O(n^2)$

#257 Explained Report Bookmark

Linked lists are best suited

- A
  for relatively permanent collections of data
- B
  for the size of the structure and the data in the structure are constantly changing
- C
  for both of above situation
- D
  for none of the above situation

Correct Answer :B

Explanation

Linked list is best suited for the size of the structure and the data in the structure are constantly changing.

From a memory allocation point of view, linked lists are more efficient than arrays. Unlike arrays, the size for a linked list is not predefined, allowing the linked list to increase or decrease in size as the program runs

#258 Explained Report Bookmark

If the values of a variable in one module is indirectly changed by another module, this situation is called

- A
  internal change
- B
  inter-module change
- C
  side effect
- D
  side-module update

Correct Answer :C

Explanation

A function, operation or statement is said to be a side effect if the value containing inside them is changed or converted by the influence of outside elements, and a value is returned to the element that had called for the operation.

#259 Explained Report Bookmark

The complexity of merge sort algorithm is

- A
  O(n)

- B
  O(log n)
- C
  O(n2)
- D
  O(n log n)

Correct Answer :D

Explanation

Merge Sort is quite fast, and has a time complexity of O(n*log n). It is also a stable sort, which means the "equal" elements are ordered in the same order in the sorted list.

#260 Explained Report Bookmark

Which of the following case does not exist in complexity theory

- A
  Best case
- B
  Worst case
- C
  Average case
- D
  Null case

Correct Answer :D

Explanation

Null case does not exist in complexity Theory

#261 Explained Report Bookmark

Two main measures of the efficiency of an algorithm are

- A
  Processor and memory
- B
  Complexity and capacity
- C
  Time and space
- D
  Data and space

Correct Answer :C

Explanation

Algorithm Efficiency. Time efficiency - a measure of amount of time for an algorithm to execute. Space efficiency - a measure of the amount of memory needed for an algorithm to execute.

#262 Explained Report Bookmark

The complexity of linear search algorithm is

- A
  O(n)
- B
  O(logn)
- C
  O(n2)
- D
  O(nlogn)

Correct Answer :A

Explanation

The worst case complexity of linear search is O(n).

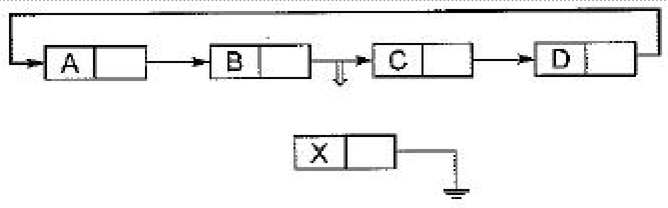
#263 Explained Report Bookmark

In a circularly linked list organization, insertion of a record involves the modification of

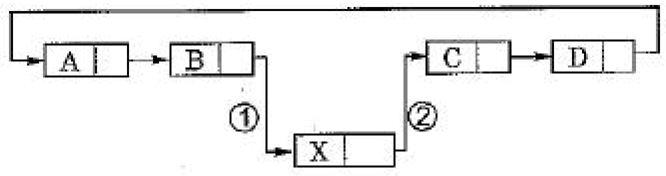
- A
  no pointer
- B
  1 pointer
- C
  2 pointers
- D
  3 pointers

Correct Answer :C

Explanation



In the above linked list a new record X wili be inserted in between B and C.



As you can see, two pointers are modified to insert X record.

#264 Not Explained Report Bookmark

The basic problem of space utilization has been removed by

null

- A
  Stack
- B
  Circular Queue
- C
  Double Ended Queue
- D
  Queue Size

Correct Answer :B

No Explanation Available

#265 Explained Report Bookmark

What additional requirement is placed on an array, so that binary search may be used to locate an entry?

- A
  The array elements must form a heap.
- B
  The array must have at least 2 entries
- C
  The array must be sorted
- D
  The array's size must be a power of two.

Correct Answer :C

Explanation

Binary search is used to find a values position in a sorted array From this definition, The Binary search can only be used on an array that has been sorted. It cannot be applied on an array that is not sorted. Therefore, the additional requirement that is placed on array so that binary search can be used for the location of an entry is found in option c, The array must be sorted.

#266 Explained Report Bookmark

Suppose c node in a linked list (using the IntNode structure with instance variables called data and link). What statement changes cursor so that it refers to the next node?

- A
  cursor++;
- B
  cursor = link;
- C
  cursor += link;
- D
  cursor = cursor->link;

Correct Answer :D

Explanation

cursor = cursor->link this statement will change the current position of the cursor.

#267 Explained Report Bookmark

What is the worst case time complexity of quick sort algorithm.

null

- A
  O(n/2)
- B
  O(2n)
- C
  O(n)
- D
  none of above

Correct Answer :D

Explanation

Worst case time complexity of quick sort algorithm is O(n^2)

#268 Explained Report Bookmark

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

Correct Answer :A

Explanation

Even though the first two elements are already sorted, bubble sort needs 4 iterations to sort the given array.

#269 Explained Report Bookmark

What is the worst case complexity of bubble sort?

- A
  O(n2)
- B
  O(n log n)

- C
  O(n)
- D
  O(log n)

Correct Answer :A

Explanation

Bubble sort is the simplest algorithm for sorting , it repeatedly swap the adjacent element and sort the list of an array

The worst case complexity of bubble sort is O(n2) and worst case occur when array is reverse sorted

#270 Explained Report Bookmark

What is the disadvantage of selection sort?

null

- A
  It requires auxiliary memory
- B
  It is not scalable
- C
  It can be used for small keys
- D
  None of the mentioned

Correct Answer :B

Explanation

Selection sort algorithm selects the smallest element from an unsorted list in each iteration and places that element at its correct position and selection sort will perform well on a short list , if the array size increases the performance decrease . It has poor efficiency for large size array

Because it required an n-squared number step to sort the array like bubble sort. That’s why it is not scalable(or we can say not stable algorithm)

#271 Explained Report Bookmark

What is the average running time of a quick sort algorithm?

null

- A
  O(N2)
- B
  O(N)
- C
  O(N log N)
- D
  O(log N)

Correct Answer :C

Explanation

Quick sort algorithm is a divide and conquer algorithm . It select a element as pivot and divide the list into two parts and the average running time is O(n log n)

#272 Explained Report Bookmark

Find the pivot element from the given input using the median-of-three partitioning method.8, 1, 4, 9, 6, 3, 5, 2, 7, 0.

null

- A
  8
- B
  7
- C
  9
- D
  6

Correct Answer :D

Explanation

The median of three partitioning methods to find pivot elements is to select the first , middle and the last index of the array and find the centre and the value of the center index will be the pivot element of the array. Here,

0 1 2 3 4 5 6 7 8
9

8 1 4 9 6 3 5 2 7 0

Center = (first+last)/ 2

0+9/2 = 4.5 it will consider the value of index 4 which is 6

#273 Explained Report Bookmark

In the following scenarios, when will you use selection sort?

null

- A
  The input is already sorted

- B
  A large file has to be sorted
- C
  Large values need to be sorted with small keys
- D
  Small values need to be sorted with large keys

Correct Answer :C

Explanation

In selection sort we select the smallest element in the given array and compare with the rest of the element and place the element at its correct position , but when we have a large size array it;s efficiency is so poor.

Selection sort will work well when we have a small size array because it can be sorted efficiently.

#274 Explained Report Bookmark

In insertion sort algorithm, how will the array elements look like after second pass?34, 8, 64, 51, 32, 21

null

- 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

Correct Answer :D

Explanation

n insertion sort we follow some step :-

1pass -> select the first element as sorted element , here assume 34 is already sorted

34	8	64	51	32	21

2pass -> pick the next element and and compare , 34 > 8 , then swap the element

8	34	64	51	32	21

So the answer is option 4

#275 Explained Report Bookmark

What is the best case and worst case complexity of ordered linear search?

null

- A
  O(nlogn), O(logn)
- B
  O(logn), O(nlogn)
- C
  O(n), O(1)
- D
  O(1), O(n)

Correct Answer :D

Explanation

In ordered / sorted linear search , if the element of the array is already sorted then in most of the case we do not need to check the complete array and it’s best case is O(1) that means the search element is at first position and the worst case is O(n) that means the search element is at last position

#276 Explained Report Bookmark

In insertion sort, the average number of comparisons required to place the 5th element into its correct position is

null

- A
  9
- B
  4
- C
  3
- D
  14

Correct Answer :C

Explanation

In insertion sort if we want to calculate the average comparisons we use (k+1)/2 . Here , the average number of comparisons required for 5th element = (5 + 1)/2 = 6/2 = 3

3 comparison is required to place the 5th element into its correct position.

#277 Explained Report Bookmark

Which of the following ways can be used to represent a graph?

null

- A
  Adjacency List and Adjacency Matrix
- B
  Incidence Matrix
- C
  Adjacency List, Adjacency Matrix as well as Incidence Matrix
- D
  None of the mentioned

Correct Answer :C

Explanation

Graphs are used to represent many real-life applications. It is used to represent networks. In graph a finite set of vertices also called as nodes.and A finite set of ordered pair of the form (u, v) called as edge

The most commonly used representation of a graph is : Adjacency matrix and adjacency list, but in graph there are other representation also exist like Incidence Matrix and incidence list

#278 Explained Report Bookmark

Evaluate Postfix expression from given infix expression. A + B * (C + D) / F + D * E

null

- A
  ABCD+*/F+DE*
- B
   AB+CD*F/+D*E
- C
  ABCD+*F/+DE*+
- D
  AB+CD*F/+DE*

Correct Answer :C

Explanation

 postfix in data structure means (operand_1 operand_2 operator)

A + B * (C + D) / F + D * E , first we solve brackets

A + B * CD+ / F + D * E , now we will go left to right and and evaluate multiplication operator

A + BCD+* / F + D * E

A + BCD+*F/ + D * E

A + BCD+*F/ + DE *

ABCD+*F/+ + DE *

ABCD+*F/+DE *+

#279 Explained Report Bookmark

Which of the below-mentioned sorting algorithms are not stable?

- A
  Merge Sort
- B
  Bubble Sort
- C
  Selection Sort
- D
  Insertion Sort

Correct Answer :C

Explanation

Selection sort algorithm selects the smallest element from an unsorted list in each iteration and places that element at its correct position and selection sort will perform well on a short list , if the array size increases the performance decrease . It has poor efficiency for large size array

Because it required an n-squared number step to sort the array.

#280Explained Report Bookmark

which of the following statements is true about a circular linked list

null

- A
  Every node in circular is successor
- B
  Time complexity to searching an element is o(n)
- C
  Time complexity to inserting a new node at the head in list is o(n)
- D
  All above mentioned.

Correct Answer :D

Explanation

In data structure circular linked list is type of linked list in which first point to the last element and last point to the first element and it form like a circle in which every node is successor , no null element and the time complexity for searching , inserting a new node and deleting is O(n) because in circular we have to traverse the list to perform all operation

#281 Explained Report Bookmark

A type of linked list in which none of the node contains NULL pointer is

null

- A
  Singly linear linked list
- B
  Singly Circular linked list
- C
  Doubly Circular linked list
- D
  Both 2 & 3

Correct Answer :D

Explanation

 In singly and doubly circular linked list none of the nodes contain NULL pointer ,because the last node points to the first and vice versa.

#282 Explained Report Bookmark

what does the following function do for a given linked list with with first node as head?

```
void function(struct node* head)

{

if(head == NULL)

    return;



function(head -> next);

printf("%d", head-> data);

}
```

- A
  prints all nodes of linked list
- B
  prints all nodes of linked list in reverse order
- C
  prints alternate node of linked list
- D
  prints alternate nodes in reverse order

Correct Answer :D

Explanation

In this program , first traverse all nodes in the linked list and then it prints the all element in reverse order. And in this program we are traversing the list first then print the element so it start print from the last element

#283 Explained Report Bookmark

Which of the following statement/s is/are right about the linked list compared with the array?

- A
  Array have better cache locality that can make them better in terms of performance
- B
  It is easy to insert and delete the elements in linked list
- C
  The size of an array has to be pre-decided whereas size of the linked list is dynamic
- D
  ALL ABOVE MENTIONED

Correct Answer :D

Explanation

All of the points mentioned are right when we talk about comparison of array and linked list , because the size of array is fixed . in array it is necessary to specify the length during declaration but linked list memory allocate at run time

#284 Explained Report Bookmark

Which sorting algorithm is used to sort a linked list in minimum time complexity?

null

- A
  Heap sort
- B
  Merge sort
- C
  Insertion sort
- D
  Quick sort

Correct Answer :B

Explanation

In linked list the element does not get continuous memory and in merge sort we divide the list and perform the sorting and again we join it . and merge sort faster because it more evenly split the list in half. Merge sort follow divide and conquer algorithm

That means first divide the problem into subproblems and solve the sub problem and join it again.

#285 Explained Report Bookmark

Underflow condition in the linked list may occur when attempting to

null

- A
  Insert a new node and no free space of it
- B
  delete nonexistence node in a list
- C
  delete a node in empty list
- D
  Insert a new in empty list

Correct Answer :C

Explanation

In data structure underflow comes into the picture when deletion is perform on the empty list and when a user perform deletion when there is no element in a list it give a message of underflow by programmer

#286 Explained Report Bookmark

What is the difference between the circular linked list to normal linked list?

null

- A
  you cannot have next pointer points to the NULL in the circular linked list.
- B
  it's faster to traverse the elements in a circular linked list.
- C
  You may or may not have the 'next' pointer point to null in the circular linked list.
- D
  All above mentioned

Correct Answer :C

Explanation

The ‘next’ pointer points to null only when the list is empty, otherwise it points to the head of the list. Every node in circular linked list can be a starting point(head).

#287 Explained Report Bookmark

Which option is false about the Doubly linked list?

- A
  We can navigate the linked list in both directions
- B
  insertion and deletion of a node takes a bit longer.
- C
  a doubly linked list requires more space than a singly linked list.
- D
  none of the mentioned.

Correct Answer :D

Explanation

A doubly linked list has two pointers ‘left’ and ‘right’ which enable it to traverse in either direction. Compared to singly liked list which has only a ‘next’ pointer, doubly linked list requires extra space to store this extra pointer. Every insertion and deletion requires manipulation of two pointers, hence it takes a bit longer time. Implementing doubly linked list involves setting both left and right pointers to correct nodes and takes more time than singly linked list.

#288 Explained Report Bookmark

The complexity of the Bubble sort algorithm is _____________.

- A
  O(n)
- B
  O(log n)
- C
  O(n2)
- D
  O(n log n)

Correct Answer :C

Explanation

Bubble sort algorithm compares the value of the first element with the immediate next element and if it is not sorted it swaps the values and it repeatedly swaps the adjacent element.

Time complexity in worst case: in worst case, the array will be in reverse sorted order and the no. of comparisons are calculated as given below

F(n) = 1 +2+3+4+——— + (n-1)

= n (n-1)/2=O (n2)

#289 Explained Report Bookmark

What is the time complexity of searching for an element in a circular linked list?

- A
  O(n log n)
- B
  O(1)
- C
  O(n)
- D
  O(n log (n-1))

Correct Answer :C

Explanation

The time complexity is O(n) , because for searching an element in circular linked list we have to traverse the list and if search is found we return the index

#290 Explained Report Bookmark

Which of the following is an example of Postfix expression?

null

- A
  * + / D A B C
- B
  (A + B) / C
- C
  A B C ^ / D E * + A C * -

- D
  A + C /(B-D)

Correct Answer :C

Explanation

In postfix expression we use operator after operand and in the first option operator is used before operand which is called prefix and in second and fourth option operators are used between the operand which is infix expression

So the option 3 is postfix expression

#291 Explained Report Bookmark

Expression in which the Operator is written after Operand is called as ________.

null

- A
  Infix
- B
  prefix
- C
  postfix
- D
  None

Correct Answer :C

Explanation

In postfix :- (operand_1 operand_2 operator) in this we write operator after the operand

Eg :- A B + , here A and B are operand and + is operator.

#292 Explained Report Bookmark

Which of the following is false about a circular linked list?

null

- A
  Every node has a successor
- B
  Time complexity of inserting a new node at the head of the list is O(1)
- C
  Time complexity for deleting the last node is O(n)
- D
  We can traverse the whole circular linked list by starting from any point

Correct Answer :B

Explanation

Time complexity of inserting a new node at the head of the list is O(n) because you have to traverse through the list to find the tail node.

#293 Explained Report Bookmark

Consider a small circular linked list. How to detect the presence of cycles in this list effectively?

null

- A
  Keep one node as head and traverse another temp node till the end to check if its ‘next points to head
- B
  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

Correct Answer :B

Explanation

Advance the pointers in such a way that the fast pointer advances two nodes at a time and slow pointer advances one node at a time and check to see if at any given instant of time if the fast pointer points to slow pointer or if the fast pointer's 'next' points to the slow pointer. This is applicable for smaller lists.

#294 Explained Report Bookmark

Which of the following application makes use of a circular linked list?

null

- A
  Undo operation in a text editor
- B
  Recursive function calls
- C
  Allocating CPU to resources
- D
  Implement Hash Tables

Correct Answer :C

Explanation

Generally, round robin fashion is employed to allocate CPU time to resources which makes use of the circular linked list data structure. Recursive function calls use stack data structure. Undo Operation in text editor uses doubly linked lists. Hash tables uses singly linked lists.

#295 Explained Report Bookmark

What differentiates a circular linked list from a normal linked list?

null

- A
  You cannot have the ‘next’ pointer point to null in a circular linked list
- B
  It is faster to traverse the circular linked list
- C
  You may or may not have the ‘next’ pointer point to null in a circular linked list
- D
  Head node is known in circular linked list

Correct Answer :C

Explanation

The ‘next’ pointer points to null only when the list is empty, otherwise it points to the head of the list. Every node in circular linked list can be a starting point(head).

#296 Explained Report Bookmark

Which of the following real world scenarios would you associate with a stack data structure?

null

- A
  piling up of chairs one above the other
- B
  people standing in a line to be serviced at a counter

- C
  offer services based on the priority of the customer
- D
  tatkal Ticket Booking in IRCTC

Correct Answer :A

Explanation

Stack follows Last In First Out (LIFO) policy. Piling up of chairs one above the other is based on LIFO, people standing in a line is a queue and if the service is based on priority, then it can be associated with a priority queue. Tatkal Ticket Booking Follows First in First Out Policy. People who click the book now first will enter the booking page first.

#297 Explained Report Bookmark

What does ‘stack underflow’ refer to?

- A
  removing items from an empty stack
- B
  adding items to a full stack
- C
  accessing item from an undefined stack
- D
  index out of bounds exception

Correct Answer :A

Explanation

Removing items from an empty stack is termed as stack underflow.

#298 Explained Report Bookmark

Array implementation of Stack is not dynamic, which of the following statements supports this argument?

null

- A
  space allocation for array is fixed and cannot be changed during run-time
- B
  user unable to give the input for stack operations
- C
  a runtime exception halts execution
- D
  improper program compilation

Correct Answer :A

Explanation

You cannot modify the size of an array once the memory has been allocated, adding fewer elements than the array size would cause wastage of space, and adding more elements than the array size at run time would cause Stack Overflow.

#299 Explained Report Bookmark

Which of the following array element will return the top-of-the-stack-element for a stack of size N elements(capacity of stack > N).

null

- A
  S[N-1]
- B
  S[N]
- C
  S[N-2]
- D
  S[N+1]

Correct Answer :A

Explanation

Array indexing start from 0, hence N-1 is the last index

#300Explained Report Bookmark

Small value returned by the hash function is called as

null

- A
  Hash value
- B
  Hash code
- C
  Digest
- D
  Hash

Correct Answer :A

Explanation

Small value returned by the hash function or output of the hash function is often known as hash value, hash code, hash sum, checksum, digest or simply hash.

#301Explained Report Bookmark

The time complexity of an algorithm/program is

null

- A
  only execution time of a program

- B
  only compilation time of a program
- C
  execution time + compilation time
- D
  none of above

Correct Answer :A

Explanation

Time complexity of an algorithm quantifies the amount of time taken by an algorithm to run as a function of the length of the input.

#302 Explained Report Bookmark

The concatenation of two lists can be performed in O(1) time. Which of the following variation of linked list can be used?

null

- A
  Singly linked list
- B
  Doubly linked list
- C
  Circular doubly linked
- D
  Array implementation of list

Correct Answer :C

Explanation

With the help of circular doubly linked list, two lists can be concatenated in O(1) time.

#303 Explained Report Bookmark

Assume that IntQueue is an integer queue. What does the function fun do?

```
void fun(int n)

{

    IntQueue q = new IntQueue();

    q.enqueue(0);

    q.enqueue(1);

    for (int i = 0; i < n; i++)

    {

        int a = q.dequeue();

        int b = q.dequeue();

        q.enqueue(b);

        q.enqueue(a + b);

        ptint(a);
```

- A
  Prints numbers from 0 to n-1

- B
  Prints numbers from n-1 to 0
- C
  Prints first n Fibonacci numbers
- D
  Prints first n Fibonacci numbers in reverse order.

Correct Answer :B

Explanation

In this program n fibonacci number is printing , here we enqueue 0 and 1 in the queue and then we start the execution of a loop which starts from zero . and after checking the condition it enters in the loop and assign 0 to a and 1 to b (by dequeue the both element from the queue) in its 1st iteration the again enqueue the value of b (1) in the queue and enqueue the sum of a and (0+1 = 1) and print the value of a which is 0 .

Same process repeat in the next iteration and we get first n fibonacci series which is the sum of previous two elements

0 1 1 2 3 5 8…….

#304 Explained Report Bookmark

Following is C like pseudo code of a function that takes a Queue as an argument, and uses a stack S to do processing.

```
void fun(Queue *Q)

{

    Stack S;

    while (!isEmpty(Q))
    {

        push(&S, deQueue(Q));

    }



    while (!isEmpty(&S))

    {

        enQueue(Q, pop(&S));

    }

}
```

- A
  Removes the last from Q
- B
  Keeps the Q same as it was before the call
- C
  Makes Q empty
- D
  Reverses the Q

Correct Answer :D

Explanation

In this function we pass the queue and an empty stack is created . in first while loop it will execute until we get the empty queue , and we just push all element one by one from queue to empty stack , suppose we have 4 element in queue :- 1 2 3 4

Queue Q

1	2	3	4

Generally , a queue dequeue operation is performed from the starting point (called as front) and pushed in the stack one by one in every iteration . now stack look like

4

3

2

1

In the 2nd while loop , we just use the above stack and pop all the elements one by one in every iteration and enqueue in the queue . In stack all operation perform from one end which is top and then the queue look like this :-

4	3	2	1

It just reverse all the element

#305 Explained Report Bookmark

Choose the correct output for the following sequence of operations.

```
push(5)

push(8)

pop()

push(2)

push(5)

pop()

pop()

pop()

push(1)

pop()
```

- A
  8 5 2 5 1
- B
  8 5 5 2 1
- C
  8 2 5 5 1
- D
  8 1 2 5 5

Correct Answer :A

Explanation

In stack push means to insert the element in the list and pop means to delete the element

1st step :- push(5) ---> 5

2nd step :- push(8) --> 5 ,8 [Here we get 8]

3rd step :- pop () [It will pop the last inserted element which is 8 , in stack all operation performed from one end] --> 5

4th step :- push(2) --> 5,2

5th step :- push(5)-->5 , 2, 5

6th step :- pop() -->5, 2 [Now we have 8 5]

7th step :- pop-->5 [Now , 8 5 2]

8th step :- pop-> no element in stack Now it's 8 5 2 5

9th step :- push(1) --> 1

10th step :- pop --> no element in stack Final output is:-   8 5 2 5 1

#306Explained Report Bookmark

The term Push and Pop is related to

- A
  Queue
- B
  Stack
- C
  Both
- D
  None

Correct Answer :B

Explanation

the term push and pop are the two operations of stack, push means to insert the element in the stack and pop means to delete the element from the stack . In stack all operation performed from one end which is called as top

#307Explained Report Bookmark

ata Structure required to evaluate infix to postfix is __________.

null

- A
  Queue
- B
  Stack

- C
  Heap
- D
  Pointer

Correct Answer :B

Explanation

To evaluate the infix to postfix we usually use stack data structure.

Here, we have Postfix[ ] Array, Stack, Infix[ ] Array

1. First it will Scan the expression (Infix) from Left to Right [scan one character at a time].
2. Check whether the scanned character is an operator or operand.
3. If it is an operand, then copy it in the Prefix Array.
4. If it is a operator then,

- Check whether, Stack is empty or not
- If it is empty then, push the operator in the stack and go to step 5.
- If Stack is not empty then compare the precedence of Top of stack with operator.
- If the Top of Stack has higher or equal precedence then pop it and copy in the postfix array.
- If the operator has higher precedence than, push it in the stack and go to step 5.

5. It continues solving the expression until the expression comes to end.
6. Pop the remaining operand in the stack and copy it to the postfix array.

#308 Explained Report Bookmark

A linear collection of data elements where the linear node is given by means of pointer is called ________________.

null

- A
  node list
- B
  linked list
- C
  self referential list
- D
  primitive list

Correct Answer :B

Explanation

A data structure for storing collection of data and all the successive elements are connected by pointer and each element called as node and the it have two item :- first is for data and the other is for storing address of next element

#309 Explained Report Bookmark

To search for an element in a sorted array, which searching technique can be used?

null

- A
  Linear Search
- B
  Jump Search
- C
  Fibonacci Search
- D
  Binary search

Correct Answer :D

Explanation

Binary search work on divide and conquer technique . In this technique the list of arrays is divided into two parts, the left side has a smaller number from the middle element and the right side has a larger number from the middle element. And binary search is used to search the element in a sorted array

In the binary search first we divide the list into two, for that we have to find middle element

middle= (first + last) / 2, this will divide the list and in this we have 3 condition

1st:- It will compare the key/search element with middle element, if it matched the return the index of middle

2nd:- If the key element is smaller than the middle element, then it come at left side and again divide the list into two and repeat the process

3rd:- If the key element is larger than the middle element, then it come at right side of the list and repeat the process

That's why binary search is used for sorted array

#310 Explained Report Bookmark

Binary search algorithms cannot be applied to__________.

null

- A
  Sorted linked list
- B
  Sorted binary trees

- C
  Sorted binary trees
- D
  Pointer array

Correct Answer :A

Explanation

The binary search algorithm cannot be applied to sorted linked list. Binary search algorithm is based on the logic of reducing your input size by half in every step until your search succeeds or input gets exhausted but in case of Linked list you don't have indexes to access items. To perform any operation on a list item, you first have to reach it by traversing all items before it. So to divide list by half you first have to reach middle of the list then perform a comparison.

#311 Explained Report Bookmark

A binary search tree contains values 7, 8, 13, 26, 35, 40, 70, 75. Which one of the following is a valid post-order sequence of the tree provided the pre-order sequence as 35, 13, 7, 8, 26, 70, 40 and 75?

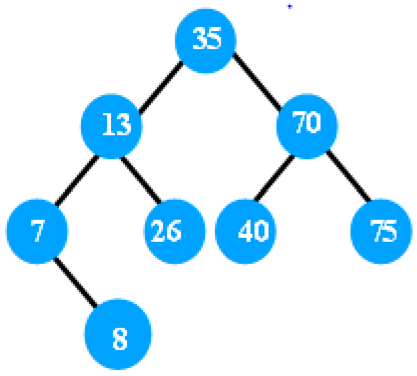
null

- A
  7, 8, 26, 13, 75, 40, 70, 35
- B
  26, 13, 7, 8, 70, 75, 40, 35
- C
  7, 8, 13, 26, 35, 40, 70, 75
- D
  8, 7, 26, 13, 40, 75, 70, 35

Correct Answer :D

Explanation

The binary tree contains values 7, 8, 13, 26, 35, 40, 70, 75. The given pre-order sequence is 35, 13, 7, 8, 26, 70, 40, and 75. So, the binary search tree formed is



Thus post-order sequence for the tree is 8, 7, 26, 13, 40, 75, 70 and 35.

#312 Explained Report Bookmark

How many stacks are needed to implement a queue. Consider the situation where no other data structure like arrays, linked list is available to you.

- A 1
- B 2
- C 3
- D 4

Correct Answer :B

Explanation

A queue can be implemented using two stacks. See following for implementation.

#313 Explained Report Bookmark

Which of the following sorting algorithms can be used to sort a random linked list with minimum time complexity?

- AInsertion Sort
- BQuick Sort
- CHeap Sort
- DMerge Sort

Correct Answer :D

Explanation

Both Merge sort and Insertion sort can be used for linked lists. The slow random-access performance of a linked list makes other algorithms (such as quicksort) perform poorly, and others (such as heapsort) completely impossible. Since worst case time complexity of Merge Sort is O(nLogn) and Insertion sort is O(n^2), merge sort is preferred. See following for implementation of merge sort using Linked List.

#314 **Explained** **Report** **Bookmark**

Level of a node is distance from root to that node. For example, level of root is 1 and levels of left and right children of root is 2. The maximum number of nodes on level i of a binary tree is

In the following answers, the operator '^' indicates power.

- A2^(i)-1
- B2^i
- C2^(i+1)
- D2^[(i+1)/2]

Correct Answer :A

Explanation

Number of nodes of binary tree will be maximum only when tree is full complete, therefore answer is 2^(i)-1 So, option (A) is true.

#315 Explained Report Bookmark

Which one of the following is an application of Stack Data Structure?

- A Managing function calls
- B The stock span problem
- C Arithmetic expression evaluation
- D All of the above

Correct Answer :D

Explanation

all

#316 Explained Report Bookmark

Which of the following is a true about Binary Trees

- A Every binary tree is either complete or full.
- B Every complete binary tree is also a full binary tree
- C Every full binary tree is also a complete binary tree.
- D None of the above

Correct Answer :D

Explanation

A full binary tree (sometimes proper binary tree or 2-tree or strictly binary tree) is a tree in which every node other than the leaves has two children. A complete binary tree is a binary tree in which every level, except possibly the last, is completely filled, and all nodes are as far left as possible.

#317 Explained Report Bookmark

Which of the following traversal outputs the data in sorted order in a BST?

- A Preorder
- B Level order
- C Inorder
- D Postorder

Correct Answer :C

Explanation

Inorder traversal of a BST outputs data in sorted order

#318 Explained Report Bookmark

We are given a set of n distinct elements and an unlabeled binary tree with n nodes. In how many ways can we populate the tree with the given set so that it becomes a binary search tree?

- A 0
- B 1
- C (1/(n+1)).2nCn
- D n!

Correct Answer :B

Explanation

There is only one way. The minimum value has to go to the leftmost node and the maximum value to the rightmost node. Recursively, we can define for other nodes.

#319 Explained Report Bookmark

Which of the following points is/are true about Linked List data structure when it is compared with array

- AArrays have better cache locality that can make them better in terms of performance.
- BIt is easy to insert and delete elements in Linked List
- CThe size of array has to be pre-decided, linked lists can change their size any time.
- DAll of the above

Correct Answer :D

Explanation

Go through Linked List vs Array

#320 Explained Report Bookmark

Which of the following operations is not O(1) for an array of sorted data. You may assume that array elements are distinct.

- AFind the ith largest element
- BDelete an element
- CFind the ith smallest element
- DAll of the above

Correct Answer :B

Explanation

The worst case time complexity for deleting an element from array can become O(n).

#321 Explained Report Bookmark

Which of the following is true about linked list implementation of queue?

- AIn push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from end.
- BIn push operation, if new nodes are inserted at the end, then in pop operation, nodes must be removed from the beginning.
- CBoth of the above
- DNone of the above

Correct Answer :C

Explanation

To keep the First In First Out order, a queue can be implemented using linked list in any of the given two ways.

#322 Explained Report Bookmark

The number of leaf nodes in a rooted tree of n nodes, with each node having 0 or 3 children is:

- An/2
- B(n-1)/3
- C(n-1)/2
- D(2n+1)/3

Correct Answer :D

Explanation

Let L be the number of leaf nodes and I be the number of internal nodes, then following relation holds for above given tree (For details, please see question 3 of this post) L = (3-1)I + 1 = 2I + 1 Total number of nodes(n) is sum of leaf nodes and internal nodes n = L + I After solving above two, we get L = (2n+1)/3

#323 Explained Report Bookmark

The preorder traversal sequence of a binary search tree is 30, 20, 10, 15, 25, 23, 39, 35, 42. Which one of the following is the postorder traversal sequence of the same tree?

- A 10, 20, 15, 23, 25, 35, 42, 39, 30
- B 15, 10, 25, 23, 20, 42, 35, 39, 30
- C 15, 20, 10, 23, 25, 42, 35, 39, 30
- D 15, 10, 23, 25, 20, 35, 42, 39, 30

Correct Answer :D

Explanation

The following is the constructed tree 30 / \ 20 39 / \ / \ 10 25 35 42 \ / 15 23

#324 Explained Report Bookmark

Level of a node is distance from root to that node. For example, level of root is 1 and levels of left and right children of root is 2. The maximum number of nodes on level i of a binary tree is

In the following answers, the operator '^' indicates power.

- A 2^(i)-1
- B 2^i
- C 2^(i+1)
- D 2^[(i+1)/2]

Correct Answer :A

Explanation

Number of nodes of binary tree will be maximum only when tree is full complete, therefore answer is 2^(i)-1 So, option (A) is true.

#325 Explained Report Bookmark

The number of structurally different possible binary trees with 4 nodes is

- A 14
- B 12
- C 36
- D 16

Correct Answer :A

Explanation

The total number of structurally different possible binary trees can be found out using the Catalon number which is (2n)!/ (n! *(n+1)!). Here n=4, so, answer is 14

#326 Explained Report Bookmark

Which of the following sequences denotes the post order traversal sequence of the given tree?

```
     a

    /   \

   b     e

  / \   /

 c  d  f

/
```

g

- A
  f e g c d b a
- B
  g c b d a f e
- C
  g c d b f e a
- D
  f e d g c b a

Correct Answer :C

Explanation

post order traversal can shortly be explained by 3 characters. LRN where L is left, R is right and N is node. We firstly visit the left node of level n of the left side first i.e. last level. If no left node is available then mark the right node if there's any. And then we mark the root.

1. Here g is the left node of last level
2. Since there is no right node, we mark node c.
3. Since the c is left node that we have visited, we next mark d which is right node
4. Then b and so on.
5. At the end when we reach level 0, it only and only has a root node which must be marked at the end. So we explore the right side of the tree.
6. We repeat the same process as we have performed on the left side of the tree.

#327Explained Report Bookmark

The maximum number of binary trees that can be formed with three unlabeled nodes is:

- A
  1

- B
  3
- C
  4
- D
  5

Correct Answer :D

Explanation

he number of unlabeled Binary Tree with n nodes is equal to the number of Binary Search Trees with n nodes.

BST(n) = C(2*n , n) / (n+1)

Therefore Unlabeled BT(n) = C(2*n , n) / (n+1)

Therefore number of unlabeled BT with 3 nodes is given by:—

=> C(2*3, 3) /(3+1)

=> C(6,3)/4

=> 6!/(3!*3!*4)

=> 6*5/6

=> 5

So the number of unlabeled Binary Tree with 3 nodes is 5.

#328 Explained Report Bookmark

Which of the following sorting algorithms can be used to sort a random linked list with minimum time complexity?

- AInsertion Sort
- BQuick Sort
- CHeap Sort
- DMerge Sort

Correct Answer :D

Explanation

Both Merge sort and Insertion sort can be used for linked lists. The slow random-access performance of a linked list makes other algorithms (such as quicksort) perform poorly, and others (such as heapsort) completely impossible. Since worst case time complexity of Merge Sort is O(nLogn) and Insertion sort is O(n^2), merge sort is preferred. See following for implementation of merge sort using Linked List

#329Explained Report Bookmark

Let S be a stack of size n >= 1. Starting with the empty stack, suppose we push the first n natural numbers in sequence, and then perform n pop operations. Assume that Push and Pop operation take X seconds each, and Y seconds elapse between the end of one such stack operation and the start of the next operation. For m >= 1, define the stack-life of m as the time elapsed from the end of Push(m) to the start of the pop operation that removes m from S. The average stack-life of an element of this stack is

- A
  n(X+ Y)
- B
  3Y + 2X
- C
  n(X + Y)-X
- D
  Y + 2X

Correct Answer :C

Explanation

https://youtu.be/Cq7B44FFXp4

#330 Explained Report Bookmark

A strictly binary tree with 10 leaves

- A
  cannot have more than 19 nodes
- B
  has exactly 19 nodes
- C
  has exactly 17 nodes
- D
  has exactly 20 nodes

Correct Answer :B

Explanation

A strict binary tree with ‘n’ leaf nodes always have ‘2n-1’ intermediate nodes. With 10 leaf nodes a strict binary tree will have exactly 19 nodes. So, option (B) is correct.

#331 Explained Report Bookmark

Which of the following traversal outputs the data in sorted order in a BST?

- APreorder
- BInorder
- CPostorder
- DLevel order

Correct Answer :B

Explanation

Inorder traversal of a BST outputs data in sorted order. Read here for details.

#332 Explained Report Bookmark

A mathematical-model with a collection of operations defined on that model is called

- A
  Data Structure
- B
  Abstract Data Type
- C
  Primitive Data Type
- D
  Algorithm

Correct Answer :B

Explanation

A mathematical-model with a collection of operations defined on that model is called abstract data type

#333 Explained Report Bookmark

What are the disadvantages of arrays?

- AData structure like queue or stack cannot be implemented
- BThere are chances of wastage of memory space if elements inserted in an array are lesser than the allocated size
- CIndex value of an array can be negative
- DElements are sequentially accessed

Correct Answer :B

Explanation

Arrays are of fixed size. If we insert elements less than the allocated size, unoccupied positions can’t be used again. Wastage will occur in memory.

#334 Explained Report Bookmark

Which of the following traversals is sufficient to construct BST from given traversals 1) Inorder 2) Preorder 3) Postorder

- A Any one of the given three traversals is sufficient
- B Either 2 or 3 is sufficient
- C 2 and 3
- D 1 and 3

Correct Answer :B

Explanation

When we know either preorder or postorder traversal, we can construct the BST. Note that we can always sort the given traversal and get the inorder traversal. Inorder traversal of BST is always sorted.

#335 Explained Report Bookmark

The following numbers are inserted into an empty binary search tree in the given order: 10, 1, 3, 5, 15, 12, 16. What is the height of the binary search tree (the height is the maximum distance of a leaf node from the root)?

- A
  2
- B
  3

- C
  4
- D
  6

Correct Answer :B

Explanation

Constructed binary search tree will be..

```
                    10

                  /      \

                1        15

                \       /   \

                  3     12    16

                     \

                       5
```

```
The height of the leaf node (5) is high 3.
```

#336 Explained Report Bookmark

Which of the following is a true about Binary Trees

- A Every binary tree is either complete or full
- B Every complete binary tree is also a full binary tree
- C Every full binary tree is also a complete binary tree.
- D None of the above

Correct Answer :D

Explanation

A full binary tree (sometimes proper binary tree or 2-tree or strictly binary tree) is a tree in which every node other than the leaves has two children. A complete binary tree is a binary tree in which every level, except possibly the last, is completely filled, and all nodes are as far left as possible

#337 Explained Report Bookmark

Which of the following traversal outputs the data in sorted order in a BST?

- A Preorder
- B Inorder
- C Postorder
- D Level order

Correct Answer :B

Explanation

Inorder traversal of a BST outputs data in sorted order. Read here for details.

#338 Explained Report Bookmark

The concatenation of two lists is to be performed in O(1) time. Which of the following implementations of a list should be used?

- A singly linked list
- B doubly linked list
- C circular doubly linked list
- D array implementation of lists

Correct Answer :C

Explanation

Singly linked list cannot be answer because we cannot find last element of a singly linked list in O(1) time. Doubly linked list cannot also not be answer because of the same reason as singly linked list.

#339 Explained Report Bookmark

Which of these best describes an array?

- A A data structure that shows a hierarchical behaviour
- B Container of objects of similar types
- C Arrays are immutable once initialised
- D Array is not a data structure

Correct Answer :B

Explanation

Array contains elements only of the same type.

#340 Explained Report Bookmark

In the worst case, the number of comparisons needed to search a singly linked list of length n for a given element is

- A log 2 n
- B n/2
- C log 2 n – 1
- D n

Correct Answer :D

Explanation

In the worst case, the element to be searched has to be compared with all elements of linked list.

#341Explained Report Bookmark

Representation of data structure in memory is known as:

- A
  recursive
- B
  abstract data type
- C
  storage structure
- D
  file structure

Correct Answer :B

Explanation

Abstract Data structures are mathematical models of different Data Structures ,such as linked list ,stack , queue etc. i.e class of objects whose logical behavior is defined by a set of values and a set of operations

#342Explained Report Bookmark

How do you initialize an array in C?

- Aint arr[3] = (1,2,3);
- Bint arr(3) = {1,2,3};
- Cint arr[3] = {1,2,3};
- Dint arr(3) = (1,2,3);

Correct Answer :C

Explanation

This is the syntax to initialize an array in C.

#343 Explained Report Bookmark

A circularly linked list is used to represent a Queue. A single variable p is used to access the Queue. To which node should p point such that both the operations enQueue and deQueue can be performed in constant time?

- A rear node
- B front node
- C not possible with a single pointer
- D node next to front

Correct Answer :A

Explanation

Answer is not "(b) front node", as we can not get rear from front in O(1), but if p is rear we can implement both enQueue and deQueue in O(1) because from rear we can get front in O(1). Below are sample functions. Note that these functions are just sample are not working. Code to handle base cases is missing. [sourcecode language="C"] /* p is pointer to address of rear (double pointer). This function adds new node after rear and updates rear which is *p to point to new node */ void enQueue(struct node **p, struct node *new_node) { /* Missing code to handle base cases like *p is NULL */ new_node->next = (*p)->next; (*p)->next = new_node; (*p) = new_node /* new is now rear */ /* Note that p->next is again front and p is rear */ } /* p is pointer to rear. This function removes the front element and returns the new front */ struct node *deQueue(struct node *p) { /* Missing code to handle base cases like p is NULL, p->next is NULL,... etc */ struct node *temp = p->next->next; p->next = p->next->next; return temp; /* Note that p->next is again front and p is rear */ }[/sourcecode]

#344 Explained Report Bookmark

In a binary tree, for every node the difference between the number of nodes in the left and right subtrees is at most 2. If the height of the tree is h > 0, then the minimum number of nodes in the tree is:

- **A**
  2^{h-1}
- **B**
  $2^{h-1} + 1$
- **C**
  $2^h - 1$
- **D**
  2^h

Correct Answer :B

Explanation

```
Let there be n(h) nodes at height h.



In a perfect tree where every node has exactly

two children, except leaves, following recurrence holds.



n(h) = 2*n(h-1) + 1



In given case, the numbers of nodes are two less, therefore
```

```
n(h) = 2*n(h-1) + 1 - 2

     = 2*n(h-1) - 1
```

Now if try all options, only option (b) satisfies above recurrence.

Let us see option (B)

n(h) = $2^{h-1} + 1$

So if we substitute

n(h-1) = $2^{h-2} + 1$, we should get n(h) = $2^{h-1} + 1$

n(h) = 2*n(h-1) - 1

= 2*(2^{h-2} + 1) -1

= 2^{h-1} + 1.

#345 Explained Report Bookmark

Suppose each set is represented as a linked list with elements in arbitrary order. Which of the operations among union, intersection, membership, cardinality will be the slowest?

- A union only

- Bintersection, membership
- Cmembership, cardinality
- Dunion, intersection

Correct Answer :D

Explanation

For getting intersection of L1 and L2, search for each element of L1 in L2 and print the elements we find in L2. There can be many ways for getting union of L1 and L2. One of them is as follows a) Print all the nodes of L1 and print only those which are not present in L2. b) Print nodes of L2. All of these methods will require more operations than intersection as we have to process intersection node plus other nodes.

#346Explained Report Bookmark

You are given pointers to first and last nodes of a singly linked list, which of the following operations are dependent on the length of the linked list?

- A
  Delete the first element
- B
  Insert a new element as a first element
- C
  Delete the last element of the list
- D
  Add a new element at the end of the list

Correct Answer :C

Explanation

a) Can be done in O(1) time by deleting memory and changing the first pointer.

b) Can be done in O(1) time, see push() here

c) Delete the last element requires pointer to previous of last, which can only be obtained by traversing the list.

d) Can be done in O(1) by changing next of last and then last.

#347 Explained Report Bookmark

If arity of operators is fixed, then which of the following notations can be used to parse expressions without parentheses?

a) Infix Notation (Inorder traversal of a expression tree)

b) Postfix Notation (Postorder traversal of a expression tree)

c) Prefix Notation (Preorder traversal of a expression tree)

- A
  b and c
- B
  Only b
- C
  a, b and c
- D
  None of them

Correct Answer :A

Explanation

Both preorder and post order parse the expression tree correctly without worrying about parentheses.

For example,

Infix:

a+b*c+d*e+f+g

its prefix is

++a*bc++*defg

And postfix expression is

abc*++de*f+g++

Here prefix and postfix can be calculated in the right way by using algorithms.But inorder can give different values.

#348 Explained Report Bookmark

What are the time complexities of finding 8th element from beginning and 8th element from end in a singly linked list? Let n be the number of nodes in linked list, you may assume that n > 8.

- A O(1) and O(n)
- B O(1) and O(1)
- C O(n) and O(1)
- D O(n) and O(n)

Correct Answer :B

Explanation

Finding 8th element from beginning requires 8 nodes to be traversed which takes constant time. Finding 8th from end requires the complete list to be traversed.

#349 Explained Report Bookmark

Which one of the following is an application of Queue Data Structure?

- A
  When a resource is shared among multiple consumers
- B
  When data is transferred asynchronously (data not necessarily received at same rate as sent) between two processes
- C
  Load Balancing
- D
  All of the above

Correct Answer :D

Explanation

1) When a resource is shared among multiple consumers. Examples include CPU scheduling, Disk Scheduling.

2) When data is transferred asynchronously (data not necessarily received at same rate as sent) between two processes. Examples include IO Buffers, pipes, file IO, etc.

3)The function of the load balancer is to distribute the load on all available backends equally.

Assuming load balancer has details of load of all backends, now if new request comes it should be directed to the backend with minimum load and this can be done in O(1) using min Heap data structure.

1. Request arrives at load balancer.

2. Load balancer picks the minimum load backed and route the request to it.

3. Load balancer increases the load on this backend server and performs heapify operation. O(logn)

#350 Explained Report Bookmark

The operation of processing each element in the list is known as, ____

- A
  Sorting
- B
  Merging
- C
  Inserting
- D
  Traversal

Correct Answer :D

Explanation

The operation of processing each element in the list is known as traversal. It refers to the process of visiting (checking and/or updating) each node in a tree data structure, exactly once.

#351 Explained Report Bookmark

How many edges are present in a Complete Graph with ' N ' Vertices?

- A
  N - 1

- B
  N - 1/2
- C
  N (N - 1) / 2
- D
  (N - 1)2

Correct Answer :C

Explanation

A simpler answer without binomials: A complete graph means that every vertex is connected with every other vertex. If you take one vertex of your graph, you therefore have n− 1n− 1 outgoing edges from that particular vertex.

Now, you have nn vertices in total, so you might be tempted to say that there are n(n− 1)n(n− 1) edges in total, n− 1n− 1 for every vertex in your graph. But this method counts every edge twice, because every edge going out from one vertex is an edge going into another vertex. Hence, you have to divide your result by 2. This leaves you with n(n− 1)/2n(n− 1)/2.

#352 Explained Report Bookmark

What is the worst case complexity of bubble sort?

- AO(nlogn)
- BO(logn)
- CO(n2)
- DO(n)

Correct Answer :C

Explanation

Bubble sort works by starting from the first element and swapping the elements if required in each iteration.

#353 Explained Report Bookmark

The time required to delete a node x from a doubly linked list having n nodes is

- A O (n)
- B O (log n)
- C O (1)
- D O (n log n)

Correct Answer :C

Explanation

The time required to delete a node x from a doubly linked list having n nodes is O(1). Doubly linked list is a sequence of elements in which every node has link to its previous node and next node. Traversing can be done in both directions and displays the contents in the whole list.

#354 Explained Report Bookmark

What are the operations that can be performed on weight balanced tree?

- A all basic operations and set intersection, set union and subset test
- B all basic operations
- C set intersection, set union and subset test
- D only insertion and deletion

Correct Answer :A

Explanation

The specialty of a weight balanced tree is a part from basic operations we can perform collective operations like set intersection, which helps in rapid prototyping in functional programming languages.

#355 Explained Report Bookmark

Which of the following is / are the Applications of a Linked List?

- A
  Polynomial
- B
  Set
- C
  Sparse matrix
- D
  All of the above

Correct Answer :D

Explanation

The main Applications of Linked Lists are:

- For representing Polynomials.

- In Dynamic Memory Management.

- In balancing parentheses.

- Representing Sparse Matrix.

#356 Explained Report Bookmark

The concatenation of two lists is to be performed in O(1) time. Which of the following implementations of a list should be used?

- Asingly linked list
- Bdoubly linked list
- Ccircular doubly linked list
- DArray implementation of lists

Correct Answer :C

Explanation

Singly linked list cannot be answer because we cannot find last element of a singly linked list in O(1) time. Doubly linked list cannot also not be answer because of the same reason as singly linked list.

#357Explained Report Bookmark

How many pointers are necessarily changed for the insertion in a Linked List?

- A
  One
- B
  two
- C
  three
- D
  five

Correct Answer :B

Explanation

This depends on whether we are inserting the new node in the middle of the list (surrounded by two nodes), or at the head or tail of the list. Insertion at the middle node will affect 4 pointers whereas at the head or tail will affect only 2 pointers

#358 Explained Report Bookmark

Which of the following sorting methods would be most suitable for sorting a list?

- A
  Bubble Sort
- B
  Insertion Sort
- C
  Selection Sort
- D
  Quick Sort

Correct Answer :A

Explanation

Bubble sort would be most suitable for sorting a list which is almost sorted. Bubble sort is a type of sorting. It is used for sorting 'n' (number of items) elements.

#359 Explained Report Bookmark

Which kind of Special Vertex is not required while treating a Tree as a Graph?

- A
  Branch
- B
  Node
- C
  Root
- D
  None of the above

Correct Answer :C

Explanation

As per the rule in the tree we are supposed to partition the nodes i.e. from 1 root to n edges and nodes. But in the graph we can notice multiple cycles. Graphs can be read from eny end whereas trees are known by their root.

In the tree each node is unique, but the root is unique of all. In the graph each node is unique but there is no greater node among them if we ignore the weights vertices carry. Thus there is no concept of root in graph.

#360 Explained Report Bookmark

An ADT is defined to be a mathematical model of a user-defined type along with the collection of all ____________ operations on that model

- A
  Cardinality
- B
  Assignment
- C
  Primitive
- D
  Structured

Correct Answer :C

Explanation

Let's take an example, Stack is an Abstract Data Type. A stack ADT can have the operations push, pop, peek. These three operations define what the type can be irrespective of the language of the implementation. So we can say, Primitive data types are a form of Abstract data type.

#361 Explained Report Bookmark

The most appropriate matching for the following pairs

```
X: Indirect addressing          1 : Loops

Y: Immediate addressing         2 : Pointers

Z: Auto decrement addressing    3: Constants
```

is

- A X-3, Y-2, Z-1
- B X-I, Y-3, Z-2
- C X-2, Y-3, Z-1
- D X-3, Y-I, Z-2

Correct Answer :C

Explanation

In Indirect addressing mode the instruction does not have the address of the data to be operated on,but the instruction points where the address is stored(it is indirectly specifying the address of memory location where the data is stored or to be stored) In immediate addressing mode the data is to be used is immediately given in instruction itself;so it deals with constant data. In Autodecrement addressing mode, Before determining the effective address, the value in the base register is decremented by the size of the data item which is to be accessed. Within a loop, this addressing mode can be used to step backwards through all the elements of an array or vector. So (C) is correct option.

#362 Explained Report Bookmark

The number of swappings needed to sort the numbers 8, 22, 7, 9, 31, 19, 5, 13 in ascending order, using bubble sort is

- A
  10
- B
  12
- C
  14
- D
  16

Correct Answer :C

Explanation

In Bubble sort, largest element moves to right. So a swapping is done, when a smaller element is found on right side.

So to count number of swaps for an element, just count number of elements on right side which are smaller than it.

Array is [8, 22, 7, 9, 31, 19, 5, 13].

For 8, number of elements on right side which are smaller : 2 (7 and 5)

For 22 : 5 (7,9,19,5,13)

For 7 : 1, for 9 : 1, for 31 : 3, for 19 : 2, for 5 : 0, for 13 : 0

Adding all these, we get : 2+5+1+1+3+2 = 14

So 14 swaps will be done.

#363 Explained Report Bookmark

B+- trees are preferred to binary trees in databases because

- A
  Disk access is much slower than memory Access
- B
  Disk capacities are greater than memory capacities
- C
  Disk data transfer rates are much less than memory data transfer rates
- D
  All of above

Correct Answer :A

Explanation

Disk access is slow and B+ Tree provides search in less number of disk hits. This is primarily because unlike binary search trees, B+ trees have very high fanout (typically on the order of 100 or more), which reduces the number of I/O operations required to find an element in the tree.

#364 Explained Report Bookmark

Which of the following data structure is not linear data structure?

- AArrays
- BLinked lists
- CBoth a and b
- DNone of the above

Correct Answer :D

Explanation

Both array and linked lists are in linear data structure concepts. Linked list is a linear data structure. It is a collection of data elements, called nodes pointing to the next node by means of a pointer. Array is a collection of variables of same type that stores the elements in the sequential manner.

#365 Explained Report Bookmark

The average successful search time taken by binary search on a sorted array of 10 items is

- A2.6
- B2.7
- C2.8
- D2.9

Correct Answer :D

Explanation

The 10 items i1, i2, ..., i10 may be arranged in a binary search trees as in Fig To get i5, the number of comparision needed is 1; for i2, it is 2; for i8 it is 2; for i1 it is 3, and so on. The average is (1+(2+2) +(3+3+3+3)+(4+4+4))/10, i.e., 2.9.

#366 Explained Report Bookmark

The data for which you are searching is called

- A
  search argument
- B
  sorting argument
- C
  detection argument
- D
  binary argument

Correct Answer :A

Explanation

The value for which we are searching is called the search argument.

#367 Explained Report Bookmark

A mathematical-model with a collection of operations defined on that model is called

- A
  Data Structure
- B
  Abstract Data Type
- C
  Primitive Data Type
- D
  Algorithm

Correct Answer :B

Explanation

A mathematical-model with a collection of operations defined on that model is called abstract data type

#368 Explained Report Bookmark

A_____search begins the search with the element that is located in the middle of the array.

- ASerial
- BRandom
- CParallel
- DBinary Search

Correct Answer :D

Explanation

Binary Search algorithm can do this because it firstly searchs the middle element. If required element is less than the middle one, then search the middle element in the less than part. If required element is greater than the middle one, then search in the greater part. So the answer is 'D'.

#369 Explained Report Bookmark

Which of the following is a collection of items into which items can be inserted arbitrarity and from which only the smallest item can be removed?

- A
  Descending priority queue
- B
  Ascending priority queue
- C
  Fifo queue
- D
  None of these

Correct Answer :B

Explanation

An ascending priority queue is a collection of items into which Items can be inserted arbitrarily and from which only the smallest item can be removed.

The priority queue is a data structure in which the intrinsic ordering of the elements does determine the result of its basic operations. An ascending priority queue is a collection of items into which items can be inserted arbitrarily and from which only the smallest item can be removed.

A priority queue is an abstract data type which is like a regular queue or stack data structure, but where additionally each element has a "priority" associated with it. While priority queues are often implemented with heaps, they are conceptually distinct from heaps.

#370 Explained Report Bookmark

Which of the following is true about linked list implementation of queue?

- A In push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from end.
- B In push operation, if new nodes are inserted at the end, then in pop operation, nodes must be removed from the beginning.
- C Both of the above
- D NONE

Correct Answer :C

Explanation

To keep the First In First Out order, a queue can be implemented using linked list in any of the given two ways.

#371 Explained Report Bookmark

A priority queue is implemented as a max-heap. Initially, it has five elements. The level-order traversal of the heap is as follows: 20, 18, 15, 13, 12 Two new elements ‘10’ and ‘17’ are inserted in the heap in that order. The level-order traversal of the heap after the insertion of the element is:

- A20, 18, 17, 15, 13, 12, 10
- B20, 18, 17, 12, 13, 10, 15
- C20, 18, 17, 10, 12, 13, 15
- D20, 18, 17, 13, 12, 10, 15

Correct Answer :D

Explanation

Initially we have

When we insert 10 and 17:

We have to maintain max-heap, so:

The level-order traversal of the heap after the insertion of the element is 20, 18, 17, 13, 12, 10, 15 So, option (D) is correct.

#372 Explained Report Bookmark

Which one of the following is an application of Stack Data Structure?

- AManaging function calls
- BArithmetic expression evaluation
- CThe stock span problem
- Dall

Correct Answer :D

Explanation

NnN

#373 Explained Report Bookmark

In a complete k-ary tree, every internal node has exactly k children or no child. The number of leaves in such a tree with n internal nodes is:

- Ank

- B(n-1)k + 1
- Cn(k - 1) + 1
- Dn(k - 1

Correct Answer :C

Explanation

For an k-ary tree where each node has k children or no children, following relation holds L = (k-1)*n + 1 Where L is the number of leaf nodes and n is the number of internal nodes. Let us see following for example

#374 Explained Report Bookmark

Suppose each set is represented as a linked list with elements in arbitrary order. Which of the operations among union, intersection, membership, cardinality will be the slowest?

- Aunion only
- Bintersection, membership
- Cmembership, cardinality
- Dunion, intersection

Correct Answer :D

Explanation

For getting intersection of L1 and L2, search for each element of L1 in L2 and print the elements we find in L2. There can be many ways for getting union of L1 and L2. One of them is as follows a) Print all the nodes of L1 and print only those which are not present in L2. b) Print nodes of L2. All of these methods will require more operations than intersection as we have to process intersection node plus other nodes.

#375 Explained Report Bookmark

Which of the following sorting algorithms can be used to sort a random linked list with minimum time complexity?

- A Insertion Sort
- B Quick Sort
- C Heap Sort
- D Merge Sort

Correct Answer :D

Explanation

Both Merge sort and Insertion sort can be used for linked lists. The slow random-access performance of a linked list makes other algorithms (such as quicksort) perform poorly, and others (such as heapsort) completely impossible. Since worst case time complexity of Merge Sort is O(nLogn) and Insertion sort is O(n^2), merge sort is preferred. See following for implementation of merge sort using Linked List

#376 Explained Report Bookmark

A complete n-ary tree is a tree in which each node has n children or no children. Let I be the number of internal nodes and L be the number of leaves in a complete n-ary tree. If L = 41, and I = 10, what is the value of n?

- A 6
- B 3

- C4
- D5

Correct Answer :D

Explanation

For an n-ary tree where each node has n children or no children, following relation holds L = (n-1)*I + 1 Where L is the number of leaf nodes and I is the number of internal nodes. Let us find out the value of n for the given data. L = 41 , I = 10 41 = 10*(n-1) + 1 (n-1) = 4 n = 5

#377 Explained Report Bookmark

What does the following function do for a given Linked List with first node as *head*?

```
void fun1(struct node* head)

{

  if(head == NULL)

    return;



  fun1(head->next);

  printf("%d  ", head->data);

}
```

- APrints all nodes of linked lists
- BPrints all nodes of linked list in reverse order
- CPrints alternate nodes of Linked List
- DPrints alternate nodes in reverse order

Correct Answer :B

Explanation

fun1() prints the given Linked List in reverse manner. For Linked List 1->2->3->4->5, fun1() prints 5->4->3->2->1.

#378Explained Report Bookmark

The following postfix expression with single digit operands is evaluated using a stack:

```
                8 2 3 ^ / 2 3 * + 5 1 * -
```

Note that ^ is the exponentiation operator. The top two elements of the stack after the first * is evaluated are:

- A6, 1
- B5, 7
- C3, 2
- D1, 5

Correct Answer :A

Explanation

The algorithm for evaluating any postfix expression is fairly straightforward: 1. While there are input tokens left o Read the next token from input. o If the token is a value + Push it onto the stack. o Otherwise, the token is an operator (operator here includes both operators, and functions). * It is known a priori that the operator takes n arguments. * If there are fewer than n values on the stack (Error) The user has not input sufficient values in the expression. * Else, Pop the top n values from the stack. * Evaluate the operator, with the values as arguments. * Push the returned results, if any, back onto the stack. 2. If there is only one value in the stack o That value is the result of the calculation. 3. If there are more values in the stack o (Error) The user input has too many values.

#379 Explained Report Bookmark

In the worst case, the number of comparisons needed to search a singly linked list of length n for a given element is

- A log 2 n
- B n/2
- C log 2 n – 1
- D n

Correct Answer :D

Explanation

In the worst case, the element to be searched has to be compared with all elements of linked list.

#380 Explained Report Bookmark

Consider a single linked list where F and L are pointers to the first and last elements respectively of the linked list. The time for performing which of the given operations depends on the length of the linked list?

- ADelete the first element of the list
- BInterchange the first two elements of the list
- CDelete the last element of the list
- DAdd an element at the end of the list

Correct Answer :C

Explanation

If F and L are pointers to the first and last elements respectively of the linked list, then: i) Deleting the first element of the list will not depend on the length of the link list as F = F->next and delete first node. ii) Interchanging the first two elements of the list will also not require the length of linked list, simply by taking a temp node, swap the two nodes of the list. iii) Deleting the last element of the list will require the length traversal of the list so as to obtain the pointer of the node previous to the last node. iv) Adding an element at the end of the list, can be done by making L->next = new node So, correct option is (C).

#381Explained Report Bookmark

The complexity of Bubble sort algorithm is

- A
  O(n)
- B
  O(log n)
- C
  O(n2)
- D
  O(n log n)

Correct Answer :C

Explanation

The following table summarizes the time complexities of bubble sort in each case-

Worst Case-

- In worst case, the outer loop runs O(n) times.
- Hence, the worst case time complexity of bubble sort is O(n x n) = $O(n^2)$.

Best Case-

- In best case, the array is already sorted but still to check, bubble sort performs O(n) comparisons.
- Hence, the best case time complexity of bubble sort is O(n).

Average Case-

- In average case, bubble sort may require (n/2) passes and O(n) comparisons for each pass.
- Hence, the average case time complexity of bubble sort is O(n/2 x n) = $\Theta(n^2)$.

	Time Complexity
Best Case	O(n)
Average Case	$\Theta(n^2)$
Worst Case	$O(n^2)$

#382 Explained Report Bookmark

A complete n-ary tree is a tree in which each node has n children or no children. Let I be the number of internal nodes and L be the number of leaves in a complete n-ary tree. If L = 41, and I = 10, what is the value of n?

- A 6
- B 3
- C 4
- D 5

Correct Answer :D

Explanation

For an n-ary tree where each node has n children or no children, following relation holds L = (n-1)*I + 1 Where L is the number of leaf nodes and I is the number of internal nodes. Let us find out the value of n for the given data. L = 41 , I = 10 41 = 10*(n-1) + 1 (n-1) = 4 n = 5

#383 Explained Report Bookmark

What is the output of following function for start pointing to first node of following linked list? 1->2->3->4->5->6

```
void fun(struct node* start)
{
  if(start == NULL)
    return;
  printf("%d  ", start->data);

  if(start->next != NULL )
    fun(start->next->next);
  printf("%d  ", start->data);
}
```

- A1 4 6 6 4 1
- B1 3 5 1 3 5
- C1 2 3 5
- D1 3 5 5 3 1

Correct Answer :D

Explanation

fun() prints alternate nodes of the given Linked List, first from head to end, and then from end to head. If Linked List has even number of nodes, then skips the last node.

#384 Explained Report Bookmark

Suppose a binary tree is constructed with n nodes, such that each node has exactly either zero or two children. The maximum height of the tree will be?

- A
  (n+1)/2
- B
  (n-1)/2
- C
  n/2 -1
- D
  (n+1)/2 -1

Correct Answer :B

Explanation

Question is about full binary tree..

If it is left or right biased than gives the maximum height...

Maximum height is(n-1/ 2) taking root at height 0.

#385 Explained Report Bookmark

The indirect change of the values of a variable in one module by another module is called

- A
  internal change
- B
  inter-module change
- C
  side effect

- D
  side-module update

Correct Answer :C

Explanation

The indirect change of the values of a variable in one module by another module is called Side effect

#386 Explained Report Bookmark

Identify the data structure which allows deletions at both ends of the list but insertion at only one end.

- A
  Input-restricted deque
- B
  Output-restricted deque
- C
  Priority queues
- D
  None of above

Correct Answer :A

Explanation

The input-restricted queue is one of the types of the Deque data structure in which deletion is allowed from both the ends but the insertion is allowed from only one end.

The output-restricted queue is one of the types of the Deque data structure in which insertion is allowed from both the ends but the deletion is allowed from only one end.

#387 Explained Report Bookmark

The concatenation of two lists is to be performed in O(1) time. Which of the following implementations of a list should be used?

- A singly linked list
- B doubly linked list
- C circular doubly linked list
- D array implementation of lists

Correct Answer :C

Explanation

Singly linked list cannot be answer because we cannot find last element of a singly linked list in O(1) time. Doubly linked list cannot also not be answer because of the same reason as singly linked list.

#388 Explained Report Bookmark

Linked lists are best suited for

- A
  relatively permanent collections of data
- B
  the size of the structure and the data in the structure are constantly changing
- C
  both of above situation
- D
  none of above situation

Correct Answer :B

Explanation

A linked list is a linear data structure, in which the elements are not stored at contiguous memory locations.

#389 Explained Report Bookmark

A BST is traversed in the following order recursively: Right, root, left The output sequence will be in

- A
  Ascending order
- B
  Descending order
- C
  Bitomic sequence
- D
  No specific order

Correct Answer :B

Explanation

Lets take an example.

Suppose there is a tree,

```
      10

     /   \

   3      7

 /   \   /   \

2  4   6  8
```

1. The preorder traversal of the tree will be, 2 3 4 10 6 7 8
2. The postorder traversal will be, 8 7 6 10 4 3 2
3. As you can see postorder is exact opposite of preorder i.e. descending order

#390 Explained Report Bookmark

hich of the following is not an advantage of trees?

- AHierarchical structure
- BFaster search
- CRouter algorithms
- DUndo/Redo operations in a notepad

Correct Answer :D

Explanation

This is an application of stack.

#391 Explained Report Bookmark

To which datastructure are skip lists similar to in terms of time complexities in worst and best cases?

- Abalanced binary search trees
- Bbinary search trees
- Cbinary trees
- Dlinked lists

Correct Answer :A

Explanation

Skip lists are similar to any randomly built binary search tree. a BST is balanced because to avoid skew tree formations in case of sequential input and hence achieve O(logn) in all 3 cases. now skip lists can gurantee that O(logn) complexity for any input.

#392 Explained Report Bookmark

Given an empty AVL tree, how would you construct AVL tree when a set of numbers are given without performing any rotations?

- A just build the tree with the given input
- B find the median of the set of elements given, make it as root and construct the tree
- C use trial and error
- D use dynamic programming to build the tree

Correct Answer :B

Explanation

Sort the given input, find the median element among them, make it as root and construct left and right subtrees with elements lesser and greater than the median element recursively. this ensures the subtrees differ only by height 1.

#393 Explained Report Bookmark

Which is the most suitable construct utilized for traversing in a Circular Linked List?

- A
  do - while
- B
  while
- C
  if
- D
  for

Correct Answer :A

Explanation

as we know circular linked list list is like a circle. Last node is pointing to the first node. Thus traversing it can be made much easier using do-while loop.

1. Initialize the temp variable to head pointer
2. while loop until the next pointer of temp becomes head

As you can see do while makes it much more easy to access the nodes in circular linked list.

#394 Explained Report Bookmark

What is the time complexity improvement of skip lists from linked lists in insertion and deletion?

- A
  O(n) to O(logn) where n is number of elements
- B
  O(n) to O(1) where n is number of elements
- C
  O(n) to O(n2) where n is number of elements
- D
  no change

Correct Answer :A

Explanation

During, insertion in cases where the node to be deleted is known only by value, the list has to be searched. Thus the time complexity becomes O(n) for the linked list.

During deletion, in linked list the pointer to previous node is possibly unknown depending on the linked list type and can be found only by traversing the list from head to the node that has a next node pointer to our to be deleted node. Thus the time complexity becomes O(logn).

#395 Explained Report Bookmark

What is the disadvantage of selection sort?

- A It requires auxiliary memory
- B It is not scalable
- C It can be used for small keys
- D None of the mentioned

Correct Answer :B

Explanation

As the input size increases, the performance of selection sort decreases.

#396 Explained Report Bookmark

What is the worst case complexity of selection sort?

- A O(nlogn)
- B O(logn)
- C O(n2)
- D O(n)

Correct Answer :C

Explanation

Selection sort creates a sub-list, LHS of the ‘min’ element is already sorted and RHS is yet to be sorted. Starting with the first element the ‘min’ element moves towards the final element.

#397Explained Report Bookmark

The nodes in a skip list may have many forward references. their number is determined

- Aprobabilistically
- Brandomly
- Csequentially
- Dorthogonally

Correct Answer :A

Explanation

The number of forward references are determined probabilistically, that is why skip list is a probabilistic algorithm.

#398Explained Report Bookmark

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

Correct Answer :A

Explanation

Even though the first two elements are already sorted, bubble sort needs 4 iterations to sort the given array.

#399 Explained Report Bookmark

What is the advantage of bubble sort over other sorting techniques?

- A It is faster
- B Consumes less memory
- C Detects whether the input is already sorted
- D All of the mentioned

Correct Answer :C

Explanation

Bubble sort is one of the simplest sorting techniques and perhaps the only advantage it has over other techniques is that it can detect whether the input is already sorted.

#400 Explained Report Bookmark

What is the ' next ' field of structure node in the Queue?

- A
  Results into the storage of queue elements.
- B
  Results into the storage of address of next node by holding the next element of queue.
- C
  Results into the memory allocation of data elements to next node.
- D
  Results into the address allocation data elements to next node.

Correct Answer :B

Explanation

In data structure node holds the data of current node. If we implement queue as a link-list, then elements are stored in node.

Example:

```
struct node

{

    int data;

    struct node * next;

};
```

#401 Explained Report Bookmark

In a binary tree you have deleted a node that have two children then it is replaced by its

- **A**
  Preorder predecessor
- **B**
  Inorder predecessor
- **C**
  Inorder successor
- **D**
  None of the above

Correct Answer :C

Explanation

In binary tree the data is originally arranged in inroder way.Thus resulting we can directing replace it by inorder successor in order to continue with the operation.

#402 Explained Report Bookmark

From where does the insertion and deletion of elements get accomplished in Queues?

- A
  Front & Rear ends respectively
- B
  Rear & Front ends respectively
- C
  Only Front ends
- D
  Only Rear ends

Correct Answer :B

Explanation

In Queues the elements are inserted from the rear end and elements are deleted from the front end.

#403 Explained Report Bookmark

Which among the below mentioned entities is / are essential for an Array Representation of a Queue?

- A
  An array to hold queue elements
- B
  A variable to hold the index of front element
- C
  A variable to hold the index of rear element

- D
  All of the above

Correct Answer :D

Explanation

For array to be Queue An array should be hold queue elements, a variable should be required to hold the index of front element, a variable should be required to hold the index of rear element.

#404 Explained Report Bookmark

When is the pop operation on Stack considered to be an error?

- A
  Only when the stack is empty
- B
  Only when the stack is full
- C
  When the stack is neither empty nor full
- D
  Cannot be predicted

Correct Answer :A

Explanation

Whenever an element is popped form stack, it must be checked whether the stack is empty or not. If it is then Stack is considered as an error.

#405 Explained Report Bookmark

Which of the following techniques represents the precise sequence of an In - Order Traversal of a Binary Tree?

- A
  Visit the Root, Traverse Left Subtree, Traverse Right Subtree
- B
  Traverse Left Subtree, Visit the Root, Traverse Right Subtree
- C
  Traverse Left Subtree, Traverse Right Subtree, Visit the Root
- D
  None Of Above

Correct Answer :B

Explanation

n case of binary trees, Inorder traversal gives nodes in non-decreasing order, so to get non-decreasing order first traverse the left sub tree then visit the root and then traverse right subtree.

#406 Explained Report Bookmark

A complete binary tree with the property that key value in any node is greater than or equal to the key values in both its children is called as.

- A
  Binary search tree
- B
  Threaded binary tree
- C
  Heap
- D
  AVL tree

Correct Answer :C

Explanation

Heap is a complete binary tree with the property that key value in any node is greater than or equal to the key values in both its children. A common use of a heap is to implement a priority queue.

#407 Explained Report Bookmark

Preorder and in order of a tree is give

Preorder ----- A B D H E C F I G J K

Inorder -------- D H B E A I F C J G K

What will be the postorder?

- A
  H D E B I F J K G C A
- B
  H D E B F I J K G C A
- C
  H D E B I F J K CG A
- D
  None of the above

Correct Answer :A

Explanation

A.preorder: RootLeftRight inorder: LeftRootRight postorder: LeftRightRoot

1. We can locate the root in preorder in first location. Also in inorder we can divide the left and right subtree by locating root i.e. left side of the root indicate the left subtree elements and right side of the root indicate the right subtree elements.

2. After root left and right node takes place, in preoder next two are B and D which can be confirmed in inroder i.e. D and H where D can be either left or root. And so on.
3. Lets create a tree of it

```
        A

      /   \

   B     C

 /  \       /  \

 D   E F  G

\    /    / \

  H I   J  K
```

So from this free we can easily create a post order traversal,

1. Left:B-D ummark D because its root
2. Right H mark
3. Root D mark
4. Since left subtree is finished, we move on upper levet .

Right E mark

1. Root B mark
2. A is root. But we are supposed to mark root after we finish left and right subtrees. Thus mode ahead with right subtree and find left element on first nth level of right subtree

C-F-I

Mark I

1. No right element, Mark F as it is a root
2. G-J mark J since its a left element
3. Right K mark
4. Root G mark
5. Root C mark
6. Root A mark

#408 Explained Report Bookmark

What should be the value of rear (end) if the queue is full (elements are completely occupied)?

- A
  1
- B
  -1
- C
  MAX + 1
- D
  MAX - 1

Correct Answer :D

Explanation

MAX - 1 should be value of rear (end) if queue is full

#409 Explained Report Bookmark

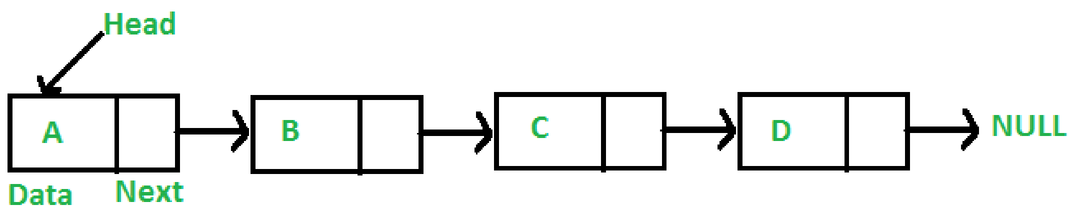
Where the elements are stored in accordance to the representation of Queue by a Linked Structure?

- A
  mesh
- B
  node
- C
  both a & b
- D
  none of the above

Correct Answer :B

Explanation

As you can see below, if we implement a queue as a link-list, then elements are stored in the node. Objects, called nodes, are linked in a linear sequence.



#410 Explained Report Bookmark

How are the elements with the same priority get processed according to the Priority Queue mechanism?

- A
  Before the processing of other elements with lower priority
- B
  After the processing of other elements with highest priority
- C
  On the basis of 'First-Come-First Served' priority

- D
  None of the Above

Correct Answer :C

Explanation

An element of higher priority is processed before any element of lower priority. And the two elements with same priority are processed according to the order in which they were added to the queue

#411Explained Report Bookmark

Which step is associated with an initialization process while converting a recursive into non - recursive algorithm?

- A
  Declaration of Stack
- B
  Pushing of all local variables and parameters into the stack
- C
  Assigning the values of formal parameters
- D
  Popping of return address

Correct Answer :A

Explanation

Any recursive function can be converted to non-recursive function through use of a stack.

#412Explained Report Bookmark

It is a set of data values and associated operations that are specified accurately, independent of any particular implementation.

- A
  Stack
- B
  ADT
- C
  Graph
- D
  Tree

Correct Answer :B

Explanation

A set of data values and associated operations that are precisely specified independent of any particular implementation. Also known as ADT.

#413 Explained Report Bookmark

A list integers is read in, one at a time, and a binary search tree is constructed. Next the tree is traversed would result in a printout which duplicates the original order of the list of integers?

- A
  Preorder
- B
  Postorder
- C
  Inorder
- D
  None of these

Correct Answer :D

Explanation

in inorder traversal the left subtree is visited first, then the root and later the right sub-tree. We should always remember that every node may represent a subtree itself. In pre order traversal the root node is visited first, then the left subtree and finally the right subtree. In post order traversal, the root node is visited last, hence the name. First we traverse the left subtree, then the right subtree and finally the root node. All three results in different ways but none fulfill the question’s demand.

#414 Explained Report Bookmark

Heap allocation is required for languages

- A
  That support recursion
- B
  That support dynamic data structure
- C
  That use dynamic scope rules
- D
  All of above

Correct Answer :B

Explanation

To use dynamic data structures we need to use heap allocation or dynamic allocation of memory.

#415 Explained Report Bookmark

Which of the following need not be a binary tree?

- A
  Search tree
- B
  Heap

- C
  AVL-Tree
- D
  B-Tree

Correct Answer :D

Explanation

In B-tree, the maximum number of child nodes a non-terminal node can have is M where M is the order of the B-tree. On the other hand, a Binary tree can have at most two subtrees or child nodes. B-tree is used when data is stored in disk whereas binary tree is used when data is stored in fast memory like RAM.. A Binary Tree is complete Binary Tree if all levels are completely filled except possibly the last level and the last level has all keys as left as possible.

#416 Explained Report Bookmark

Which of the following abstract data types can be used to represent a many to many relation?

- A
  Tree
- B
  Plex
- C
  Graph
- D
  Both (b) and (c)

Correct Answer :D

Explanation

A Graph is a non-linear data structure consisting of nodes and edges. The nodes are sometimes also referred to as vertices and the edges are lines or arcs that connect any two nodes in the graph. A plex (derived from plexus meaning any complex structure containing an intricate network of interrelated parts) consists of a set of elements called beads, where each element is an N-word vector of computer storage.

#417 Explained Report Bookmark

An algorithm is made up of 2 modules M1&M2. If order of M1 is f(n) & M2 is g(n) then the order of algorithm is?

- A
  max (f(n),g(n))
- B
  min (f(n),g(n))
- C
  f(n) + g(n)
- D
  f(n) X g(n)

Correct Answer :A

Explanation

By definition of order, there exists constants c1, ec2, n1, n2 such that

T (n) $\leq$ c1 x f (n) , for all n $\geq$ n1.

T (n) $\leq$ c2 x f (n) , for all n $\geq$ n2.

N = max (n1, n2) andC = max (c1, c2). So,

T (n) $\leq$ C x f(n) for all n $\geq$ N

T (n) $\leq$ C x g (n) for all n $\geq$ N

T(n) $\leq$ C/2 x (f(n)+g(n))

Without loss of generality, let max (f (n), g (n)) =f(n) .

So, T (n) $\leq$ C / 2 (f(n) + f (n)) $\leq$ C x f(n) .

So, order is f(n) , which is max (f(n), g (n)) , by our assumption

#418 Explained Report Bookmark

An array of n numbers is given, where n is an even number. The maximum as well as the minimum of these n numbers needs to be determined. Which of the following is TRUE about the number of comparisons needed?

- A
  At least 2n-c comparisons, for some constant c, are needed.
- B
  At most 1.5n-2 comparisons are needed.
- C
  At least nlog2n comparisons are needed.
- D
  None of the above.

Correct Answer :B

Explanation

Tournament Method Technique is one of the technique we can apply here -

finding the smallest element in the array will take n− 1 comparisons.

And finding the largest element will need-

1. After the first round , there will be exactly n/2 numbers that will pass for the largest number.
2. So the largest number should be among these 50 loosers.To find the largest number will take $n/2-1$.

Total Comparisons $=(n-1)+(n/2-1)=1.5n-2$.

#419 Explained Report Bookmark

binary tree in which if all its levels except possibly the last, have the maximum number of nodes and all the nodes at the last level appear as far left as possible, is called

- A
  Full binary tree
- B
  Binary Search Tree
- C
  Threaded tree
- D
  Complete binary tree

Correct Answer :D

Explanation

A binary tree in which if all its levels except possibly the last, have the maximum number of nodes and all the nodes at the last level appear as far as left as possible, is known as Computer binary tree.

The maximum number of nodes on level of a binary tree is : if level is 3 then there will be maximum 7 nodes in the binary tree. which is 2^3-1=8-1=7.

A binary tree is a tree data structure in which each node has at most two children, which are referred to as the left child and the right child.

Parent nodes are nodes with children, while child nodes may include references to their parents.

#420 Explained Report Bookmark

A complete binary tree with the property that the value at each node is at least as large as the values at its children is called

- A
  binary search tree
- B
  Binary Tree
- C
  Completely balanced tree
- D
  Heap

Correct Answer :D

Explanation

Heap is a complete binary tree with the property that the value of each node is at least as large as the values at its children.

Heaps are of two types:

- Min Heap
- Max Heap

The property of Min Heap is that the value of each node is either greater or equal to its parent's value and having minimum value element at the root.

The property of Max Heap is that the value of each node is equal or lesser to its parent's value and having maximum value element at the root.

#421 Explained Report Bookmark

The minimum number of comparisons required to determine if an integer appears more than n/2 times in a sorted array of n integers is

- AΘ(n)
- BΘ(logn)
- CΘ(log*n)
- DΘ(1)

Correct Answer :B

Explanation

Since it is a sorted array, we can use binary search to identify the position of the first occurence of the given integer in (logn) steps. If at all this integer repeats, its appearance has to be continuous because the array is sorted.

#422 Explained Report Bookmark

Which of the following algorithms solves the all-pair shortest path problem?

- ADijkstra's algorithm
- BFloyd's algorithm
- CPrim's algorithm
- DWarshall's algorithm

Correct Answer :B

Explanation

Dijkstra's algorithm solves single source shortest path problem. Warshall's algorithm finds transitive closure of a given graph. Prim's algorithm constructs a minimum cost spanning tree for a given weighted graph.

#423 Explained Report Bookmark

For merging two sorted lists of sizes m and n into a sorted list of size m + n, we require comparisons of

- A O(m)
- B O(n)
- C O(m+n)
- D O(log(m) + log(n))

Correct Answer :C

Explanation

Each comparison will append one item to the existing merge list. In the worst case one needs m + n - 1 comparisons which is of order m+n.

#424 Explained Report Bookmark

Which of the following sorting methods sorts a given set of items that is already in sorted order or in reverse sorted order with equal speed?

- A
  Insertion sort
- B
  Heap sort
- C
  Quick sort
- D
  Selection sort

Correct Answer :C

Explanation

Quick sort has two worst cases, when input is in either ascending or descending order, it takes same time O(n2).

#425 Explained Report Bookmark

Which of the following sorting algorithm has the worst time complexity of nlog(n)?

- A
  Insertion sort
- B
  Quick sort
- C
  Heap sort
- D
  Selection sort

Correct Answer :C

Explanation

In Heap sort, the call to procedure build_Max-heap takes O(n) time and each of O(n) calls to the function max_Heapify takes O(logn) time. So the worst case complexity of Heap sort is O(nlogn).

#426 Explained Report Bookmark

For a linear search in an array of n elements the time complexity for best, worst and average case are ...., ... and .... respectively

- A
  O(n), O(1), and O(n/2)

- B
  O(1), O(n) and O(n/2)
- C
  O(1),O(n) and O(n)
- D
  O(1), O(n) and (n-1/2)

Correct Answer :C

Explanation

Class	No. of comparisons required	Complexity
Worst-case	n	O(n)
Best-case	1	O(1)
Average-case	n/2	O(n)

Note : Asymptotic notations ignore constants so average complexity is O(n) by ignoring ½ constant

#427 Explained Report Bookmark

A binary tree having n nodes and depth d will be about complete binary tree if

- A
  any node nd at level less than d-1 has two sons
- B
  it contains log(d)+1 nodes
- C
  for any node nd in the tree with a right descendent at level d It must have a left son
- D
  all of these

Correct Answer :A

Explanation

. A binary tree is made of nodes, where each node contains a "left" reference, a "right" reference, and a data element. The topmost node in the tree is called the root. Every node in a tree is connected by a directed edge from exactly one other node. This node is called a parent.

When a tree of height d has all nodes filled from level 0 to d-1, and the leaf nodes at the dth level are filled from the left-most position, then the tree is called a complete binary tree.

#428 Explained Report Bookmark

A binary tree in which if all its levels except possibly the last, have the maximum number of nodes and all the nodes at the last level appear as far left as possible, is called

- A
  Full binary tree
- B
  Binary Search Tree

- C
  Threaded tree
- D
  Complete binary tree

Correct Answer :D

Explanation

A binary tree in which if all its levels except possibly the last, have the maximum number of nodes and all the nodes at the last level appear as far as left as possible, is known as Computer binary tree.

The maximum number of nodes on level of a binary tree is : if level is 3 then there will be maximum 7 nodes in the binary tree. which is 2^3-1=8-1=7.

A binary tree is a tree data structure in which each node has at most two children, which are referred to as the left child and the right child.

Parent nodes are nodes with children, while child nodes may include references to their parents.

#429 Explained Report Bookmark

Which of the following remarks about Tree- indexing are true?

- A
  It is an m-ary tree
- B
  Successful searches should terminate in leaf nodes
- C
  Unsuccessful searches may terminate in leaf nodes level of the tree structure
- D
  All of these

Correct Answer :D

Explanation

B+Tree indexing is a method of accessing and maintaining data. It should be used for large files that have unusual, unknown, or changing distributions

m-ary tree (also known as k-ary or k-way tree) is a rooted tree in which each node has no more than m children. A binary tree is the special case where m = 2, and a ternary tree is another case with m = 3 that limits its children to three.

B-trees are always height balanced, with all leaf nodes at the same level

#430 Explained Report Bookmark

The running time of an algorithm T(n),where 'n' is the input size, of a recursive algorithm is given as follows.is given by

T(n) =c + T(n - 1), if n > 1

d, if n ≤ 1

The order of this algorithm is

- A
  n2
- B
  n
- C
  n3
- D
  nn

Correct Answer :B

Explanation

By recursively applying the relation we finally arrive at T(n-l) = c(n-l) + T (1) = c(n-l)+d So, order is n

#431Explained Report Bookmark

The minimum number of comparisons required to determine if an integer appears more than n/2 times in a sorted array of n integers is

- AΘ(n)
- BΘ(logn)
- CΘ(log*n)
- DΘ(1)

Correct Answer :B

Explanation

Since it is a sorted array, we can use binary search to identify the position of the first occurence of the given integer in (logn) steps. If at all this integer repeats, its appearance has to be continuous because the array is sorted.

#432Explained Report Bookmark

Six files Fl, F2, F3, F4, F5 and F6 have 100,200,50,80, 120, 150 number of records respectively. In what order should they be stored so as to optimize access time? Assume each file is accessed with the same frequency.

- AF3, F4, Fl, F5, F6, F2
- BF2, F6, F5, Fl, F4, F3
- CFl, F2, F3, F4, F5, F6
- DOrdering is immaterial as all files are accessed with the same frequency

Correct Answer :A

Explanation

Since the access is sequential, greater the distance, greater will be the access time. Since all the files are referenced with equal frequency, overall access time can be reduced by arranging them as in option (a).

#433 Explained Report Bookmark

Representation of data structure in memory is known as:

- A
  Recursive
- B
  Abstract data type
- C
  Storage structure
- D
  File structure

Correct Answer :B

Explanation

Abstract Data type (ADT) is a type (or class) for objects whose behaviour is defined by a set of value and a set of operations.The definition of ADT only mentions what operations are to be performed but not how these operations will be implemented. It does not specify how data will be organized in memory and what algorithms will be used for implementing the operations. It is called “abstract” because it gives an implementation-independent view. The process of providing only the essentials and hiding the details is known as abstraction.

#434 Explained Report Bookmark

What does the following function do for a given Linked List with first node as *head*?

```
void fun1(struct node* head)

{

  if(head == NULL)

    return;



  fun1(head->next);

  printf("%d  ", head->data);

}
```

- APrints all nodes of linked lists
- BPrints all nodes of linked list in reverse order
- CPrints alternate nodes of Linked List
- DPrints alternate nodes in reverse order

Correct Answer :B

Explanation

fun1() prints the given Linked List in reverse manner. For Linked List 1->2->3->4->5, fun1() prints 5->4->3->2->1.

#435 Explained Report Bookmark

Which of the following is true about linked list implementation of stack?

- AIn push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from end
- BIn push operation, if new nodes are inserted at the end, then in pop operation, nodes must be removed from the beginning.
- CBoth
- DNone

Correct Answer :D

Explanation

To keep the Last In First Out order, a stack can be implemented using linked list in two ways: a) In push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from beginning. b) In push operation, if new nodes are inserted at the end of linked list, then in pop operation, nodes must be removed from end.

#436 Explained Report Bookmark

What is the speciality about the inorder traversal of a binary search tree?

- AIt traverses in a non increasing order
- BIt traverses in an increasing order
- CIt traverses in a random fashion
- DIt traverses based on priority of the node

Correct Answer :B

Explanation

As a binary search tree consists of elements lesser than the node to the left and the ones greater than the node to the right, an inorder traversal will give the elements in an increasing order.

#437 Explained Report Bookmark

Level of a node is distance from root to that node. For example, level of root is 1 and levels of left and right children of root is 2. The maximum number of nodes on level i of a binary tree is

In the following answers, the operator '^' indicates power.

- A2^(i)-1
- B2^i
- C2^(i+1)
- D2^[(i+1)/2]

Correct Answer :A

Explanation

Number of nodes of binary tree will be maximum only when tree is full complete, therefore answer is 2^(i)-1 So, option (A) is true.

#438 Explained Report Bookmark

An implementation of a queue Q, using two stacks S1 and S2, is given below:

```
void insert(Q, x) {

   push (S1, x);

}



void delete(Q){

   if(stack-empty(S2)) then

      if(stack-empty(S1)) then {
          print(“Q is empty”);

          return;

      }

      else while (!(stack-empty(S1))){

          x=pop(S1);

          push(S2,x);

      }

   x=pop(S2);

}
```

Let n insert and m (<=n) delete operations be performed in an arbitrary order on an empty queue Q. Let x and y be the number of push and pop operations performed respectively in the process. Which one of the following is true for all m and n?

- A n+m <= x < 2n and 2m <= y <= n+m
- B n+m <= x < 2n and 2m<= y <= 2n
- C 2m <= x < 2n and 2m <= y <= n+m
- D 2m <= x <2n and 2m <= y <= 2n

Correct Answer :A

Explanation

The order in which insert and delete operations are performed matters here. The best case: Insert and delete operations are performed alternatively. In every delete operation, 2 pop and 1 push operations are performed. So, total m+ n push (n push for insert() and m push for delete()) operations and 2m pop operations are performed. The worst case: First n elements are inserted and then m elements are deleted. In first delete operation, n + 1 pop operations and n push operation are performed. Other than first, in all delete operations, 1 pop operation is performed. So, total m + n pop operations and 2n push operations are performed (n push for insert() and n push for delete())

#439 Explained Report Bookmark

If -*+abcd = 11, find a, b, c, d using evaluation of prefix algorithm

- A a=2, b=3, c=5, d=4
- B a=1, b=2, c=5, d=4
- C a=5, b=4, c=7,d=5
- D a=1, b=2, c=3, d=4

Correct Answer :B

Explanation

The given prefix expression is evaluated as ((1+2)*5)-4 =11 while a=1, b=2, c=5, d=4.

#440 Explained Report Bookmark

What is the functionality of the following piece of code?

```
public int function (int data)

{

    Node temp = head;

    int var = 0 ;

    while(temp!=null){
        if(temp.getData() == data){
            return var ;
        }
        var = var+i;
        temp = temp.getNext();
    }
    return Integer.MIN_VALUE


}
```

- A
  Find and delete a given element in the list
- B
  Find and return the given element in the list
- C
  Find and return the position of the given element in the list
- D
  Find and insert a new element in the list

Correct Answer :C

Explanation

When temp is equal to data, the position of data is returned.

#441 Explained Report Bookmark

How do you initialize an array in C?

- Aint arr[3] = (1,2,3);
- Bint arr(3) = {1,2,3};
- Cint arr[3] = {1,2,3};
- Dint arr(3) = (1,2,3);

Correct Answer :C

Explanation

int arr[3] = {1,2,3}; This is the syntax to initialize an array in C.

#442 Explained Report Bookmark

Suppose each set is represented as a linked list with elements in arbitrary order. Which of the operations among union, intersection, membership, cardinality will be the slowest?

- Aunion only
- Bintersection, membership
- Cmembership, cardinality
- Dunion, intersection

Correct Answer :D

Explanation

For getting intersection of L1 and L2, search for each element of L1 in L2 and print the elements we find in L2. There can be many ways for getting union of L1 and L2. One of them is as follows a) Print all the nodes of L1 and print only those which are not present in L2. b) Print nodes of L2. All of these methods will require more operations than intersection as we have to process intersection node plus other nodes.

#443 Explained Report Bookmark

A function f defined on stacks of integers satisfies the following properties. f(∅) = 0 and f (push (S, i)) = max (f(S), 0) + i for all stacks S and integers i.

If a stack S contains the integers 2, -3, 2, -1, 2 in order from bottom to top, what is f(S)?

- A6
- B4
- C3
- D2

Correct Answer :C

Explanation

f(S) = 0, max(f(S), 0) = 0, i = 2 f(S)new = max(f(S), 0) + i = 0 + 2 = 2 f(S) = 2, max(f(S), 0) = 2, i = -3 f(S)new = max(f(S), 0) + i = 2 - 3 = -1 f(S) = -1, max(f(S), 0) = 0, i = 2 f(S)new = max(f(S), 0) + i = 0 + 2 = 2 f(S) = 2, max(f(S), 0) = 2, i = -1 f(S)new = max(f(S), 0) + i = 2 - 1 = 1 f(S) = 1, max(f(S), 0) = 1, i = 2 f(S)new = max(f(S), 0) + i = 1 + 2 = 3 Thus, option (C) is correct.

#444Explained Report Bookmark

We are given a set of n distinct elements and an unlabeled binary tree with n nodes. In how many ways can we populate the tree with the given set so that it becomes a binary search tree?

- A0
- B1
- Cn!
- D(1/(n+1)).2nCn

Correct Answer :B

Explanation

There is only one way. The minimum value has to go to the leftmost node and the maximum value to the rightmost node. Recursively, we can define for other nodes.

#445Explained Report Bookmark

Which if the following is/are the levels of implementation of data structure

- A
  Abstract level
- B
  Application level
- C
  Implementation level
- D
  All of the above

Correct Answer :D

Explanation

All are the levels of implementation of data structure.

#446Explained Report Bookmark

binary search tree whose left subtree and right subtree differ in height by at most 1 unit is called ……

- A
  AVL tree
- B
  Red-black tree
- C
  Lemma tree

- D
  None of the above

Correct Answer :A

Explanation

An AVL tree is the binary search tree in which every sub tree is an AVL tree but is not entirely balanced. It takes less time to calculate the height of sub trees in the AVL tree which is the main advantage of these trees. So it is widely used in computer algorithms in a design method. The various operations on AVL trees such as insertion, deletions, searching, etc.

#447 Explained Report Bookmark

………….. level is where the model becomes compatible executable code

- A
  Abstract level
- B
  Application level
- C
  Implementation level
- D
  All of the above

Correct Answer :C

Explanation

Implementation level has access to the structure including data which converts model to compatible executable code where executable code generally refers to machine language

#448 Explained Report Bookmark

Stack is also called as

- A
  Last in first out
- B
  First in last out
- C
  Last in last out
- D
  First in first out

Correct Answer :A

Explanation

A stack is a typical data structure that may be accessed using the LIFO method. In a stack, each item is placed on top of the previous item, one at a time

#449 Explained Report Bookmark

Which of the following is true about the characteristics of abstract data types?

i) It exports a type.

ii) It exports a set of operations

- A
  True, False
- B
  False, True
- C
  True, True
- D
  False, False

Correct Answer :C

Explanation

To build an ADT one must export type definition along with set of operations that will be needed in order to perform certain tasks. The data structure can only be accessed with defined operations. This set of operations is called interface and is exported by the entity. An entity with the properties just described is called an abstract data type (ADT)

#450 Explained Report Bookmark

……… is not the component of the data structure.

- A
  Operations
- B
  Storage Structures
- C
  Algorithms
- D
  None of the above

Correct Answer :D

Explanation

Data Definitions	Data definitions describe how data is entered into Primavera Unifier and stored. The definition consists of the data type, data size, and input method. Data definitions govern the behavior of data elements.

Data Elements	Data elements are the fields the users see on the forms in Primavera Unifier, such as a text box or menu. The behavior of a data element is governed by the data definition on which it is built.
Status	A status indicates the condition or state of a record, line item, or asset. Every record, line item, and asset must have a status indication at every step in Primavera Unifier.
Dynamic Data Sets	Dynamic data sets are a way of narrowing the number of possible choices a user can make. Narrowing the choices makes entering data faster and more efficient for users. Dynamic data sets dictate how fields behave when the user has multiple-choice options.
Tags	Tags work with linked elements to connect schedule sheet activities with business processes, shells, and configurable manager. Tags allow a one-to-many relationship between a data element on the master schedule sheet and multiple data elements on a form.

#451 Explained Report Bookmark

Inserting an item into the stack when stack is not full is called …………. Operation and deletion of item form the stack, when stack is not empty is called ………..operation.

- A
  push, pop
- B
  pop, push
- C
  insert, delete

- D
  delete, insert

Correct Answer :A

Explanation

A stack is a simple last-in, first-out (LIFO) data structure. That is, the last data element stored in a stack is the first data element retrieved from the stack. The common analogy is a stack of plates in a cafeteria: when you go through the line, you pop the top plate off the stack; the dish washer (stepping away from reality a bit), pushes a single clean plate on top of the stack.

push() function is used to insert an element at the top of the stack.

The element is added to the stack container and the size of the stack is increased by 1.

pop() function is used to remove an element from the top of the stack(newest element in the stack).

The element is removed to the stack container and the size of the stack is decreased by 1.

#452 Explained Report Bookmark

……… is very useful in situation when data have to stored and then retrieved in reverse order.

- A
  Stack
- B
  Queue
- C
  List

- D
  Link list

Correct Answer :A

Explanation

Concept: Basic Data Structures (Stack, Queue, Dequeue)

#453 Explained Report Bookmark

A ……. is a data structure that organizes data similar to a line in the supermarket, where the first one in line is the first one out.

- A
  Queue linked list
- B
  Stacks linked list
- C
  Both of them
- D
  Neither of them

Correct Answer :A

Explanation

A queue linked list is a data structure that organizes data similar to a line in the supermarket, where the first one in line is the first one out.

#454 Explained Report Bookmark

Which of the following is non-liner data structure?

- A
  Stacks

- B
  List
- C
  Strings
- D
  Trees

Correct Answer :D

Explanation

Examples of non-linear data structures are Tree, BST, Graphs etc. ... In linear data structure, data elements are sequentially connected and each element is traversable through a single run. In non-linear data structure, data elements are hierarchically connected and are present at various levels

#455 Explained Report Bookmark

Choose the correct statements

- A
  Internal sorting is used if the number of items to be sorted is very large
- B
  External sorting is used if the number of items to be sorted is very large
- C
  External sorting needs auxilary storage
- D
  Both (b) & (c)

Correct Answer :D

Explanation

1. Internal sorting is such a process that takes place entirely within the main memory of a computer. This is only possible whenever data items, to be

sorted, is small enough to be accommodated in main memory. Hence, internal sorting do not need auxiliary storage.

2. External sorting is just opposite to interval sorting and used when input data, to be sorted, is very large. Hence, external sorting need auxiliary storage.

#456 Not Explained Report Bookmark

The correct matching for the following pairs is

(A) All pairs shortest path	(1) Greedy
(B) Quick sort	(2) Depth-first search
(C) Minimum weight spanning tree	(3) Dynamic programming
(D) Connected Components	(4) Divide and conquer

- A A-2 , B-4 , C-1, D-3
- B A-3 , B-4 , C-l , D-2
- C A-3 , B-4 , C-2 , D-1
- D A-4 , B-1 , C-2 , D-3

Correct Answer :B

No Explanation Available

#457 Explained Report Bookmark

A complete binary tree with the property that the value at each node is at least as large as the values at its children is called

- A
  binary search tree
- B
  Binary Tree
- C
  Completely balanced tree
- D
  Heap

Correct Answer :D

Explanation

Heap is a complete binary tree with the property that the value of each node is at least as large as the values at its children.

#458 Explained Report Bookmark

The number of nodes in a complete binary tree of level 5 is

- A
  15
- B
  20
- C
  63
- D
  71

Correct Answer :C

Explanation

formula used: (2^(d+1))-1 where d is the level i.e. 5

#459 Explained Report Bookmark

A binary tree of depth "d" is an almost complete binary tree if

- A
  each leaf in the tree is either at level
- B
  for any node
- C
  both a and b
- D
  None of these

Correct Answer :C

Explanation

A binary tree of depth “d” is an almost complete binary tree if each leaf in the tree is either at level “d” or at level “d–1” and for any node “n” in the tree with a right descendent at level “d” all the left descendants of “n” that are leaves, are also at level “d”.

#460 Explained Report Bookmark

An ADT is defined to be a mathematical model of a user-defined type along with the collection of all ____________ operations on that model

- A
  Cardinality
- B
  Assignment
- C
  Primitive
- D
  Structured

Correct Answer :C

Explanation

Abstract Data type is a theoretical concept. An abstract data type (ADT) is a mathematical model for data types where a data type is defined by its behaviour (semantics) from the point of view of a user of the data, specifically in terms of possible values, possible operations on data of this type, and the behaviour of these operations.A set of data values and associated operations that are precisely specified independent of any particular implementation. Abstract data type (ADT) is a specification of a set of data and the set of operations that can be performed on the data.

Lets take an example, Stack is an Abstract Data Type. A stack ADT can have the operations push, pop, peek. These three operations define what the type can be irrespective of the language of the implementation.

#461 Explained Report Bookmark

Which of the following is true about linked list implementation of stack?

- A In push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from end
- B In push operation, if new nodes are inserted at the end, then in pop operation, nodes must be removed from the beginning.
- C Both of the above
- D None

Correct Answer :D

Explanation

To keep the Last In First Out order, a stack can be implemented using linked list in two ways: a) In push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from beginning. b) In push operation, if new nodes are inserted at the end of linked list, then in pop operation, nodes must be removed from end.

#462 Explained Report Bookmark

If each node in a tree has value greater the every value in its left subtree and has value less than every value in its right subtree, the tree is called

- A
  Complete tree
- B
  Full binary tree
- C
  Binary search tree
- D
  AVL tree

Correct Answer :C

Explanation

A binary search tree (BST) is a binary tree where every node in the left subtree is less than the root, and every node in the right subtree is of a value greater than the root. The properties of a binary search tree are recursive: if we consider any node as a “root,” these properties will remain true.

#463 Explained Report Bookmark

Which of the following statements is false ?

- A
  Every tree is a bipartite graph
- B
  A tree contains a cycle
- C
  A tree with n nodes contains n-1 edges
- D
  A tree is a connected graph

Correct Answer :B

Explanation

A tree does not contains cycle while a graph may or may not contains a cycle.

A tree with cycle is called a Graph.

#464Explained Report Bookmark

The information about an array that is used in a program will be stored in

- A
  symbol table
- B
  activation record
- C
  system table
- D
  dope vector

Correct Answer :D

Explanation

Answer would be 4) Dope Vector

In computer programming, a dope vector is a data structure used to hold information about a data object, e.g. an array, especially its memory layout.

A dope vector typically contains information about the type of array element, rank of an array, the extents of an array, and the stride of an array as well as a pointer to the block in memory containing the array elements.

#465Explained Report Bookmark

The smallest number of key that will force a B-tree of order 3 to have a height 3 is

- A
  12

- B
  10
- C
  07
- D
  none

Correct Answer :C

Explanation

Single median is based on among leaf's elements and the new element and after that the split the node into two nodes.

Median value is shifted to parent node. Split node with the two nodes after m/2-1 where m is order of tree position.

The median value is shifted for parent node and inserts value to the appropriate position.

#466 Explained Report Bookmark

For merging two sorted lists of sizes m and n into a sorted list of size m + n, we require comparisons of

- AO(m)
- BO(n)
- CO(m+n)
- DO(log(m) + log(n))

Correct Answer :C

Explanation

Each comparison will append one item to the existing merge list. In the worst case one needs m + n - 1 comparisons which is of order m+n.

#467Explained Report Bookmark

Which of the following is useful in traversing a given graph by breadth-first search?

- A
  stacks
- B
  set
- C
  List
- D
  Queue

Correct Answer :D

Explanation

BFS performs level-order traversal which can be fairly done using a queue. A queue uses FIFO ordering and the nodes that we enqueue first are explored first maintaining the order of traversal.

#468Explained Report Bookmark

The average number of comparisons performed by the merge sort algorithm, in merging two sorted liss of length 2 is

- A8/3
- B8/5
- C11/7
- D11/6

Correct Answer :A

Explanation

Merge-sort combines two given sorted lists into one sorted list. For this problem let the final sorted order be- 1, b, c, d. The two lists (of length two each) should fall into one of the following 3 categories.

(i) a, b and c,d (ii) a, c and b, d (iii) a, d and b, c

The number of comparisons needed in each case will be 2, 3, 3. So, average number of comparisons will be (2=3+3)/3= 8/3

Here is a better way of doing;

Let list L1 have the items a,c and L2 have the items b,d.

The tree drawn below, depicts the different possible cases. (a & b means a is compared with b. If a is smaller, the edge will be labeled a. The number within a circle, beside the leaf nodes, is the number of comparisons, needed to reach it.)

From the tree, we find there are 6 possible ways, total number of comparisons needed is 3+3+2+2+3+3=16. So, average number of comparisons is 16/6= 8/3.

#469 Explained Report Bookmark

In a heap, element with the greatest key is always in the ___________ node

- A
  leaf

- B
  root
- C
  first node of left sub tree
- D
  first node of right sub tree

Correct Answer :B

Explanation

In a heap, element with the greatest key is always in the root node. Heap is a complete binary tree such that root node is always greater than or equal to the key value in both its children.

#470 Explained Report Bookmark

The number of swappings needed to sort the numbers 8, 22, 7, 9, 31, 19, 5, 13 in ascending order, using bubble sort is

- A
  11
- B
  12
- C
  13
- D
  14

Correct Answer :D

Explanation

In Bubble sort, largest element moves to right. So a swapping is done, when a smaller element is found on right side.

So to count number of swaps for an element, just count number of elements on right side which are smaller than it.

Array is [8, 22, 7, 9, 31, 19, 5, 13].

For 8, number of elements on right side which are smaller : 2 (7 and 5)

For 22 : 5 (7,9,19,5,13)

For 7 : 1, for 9 : 1, for 31 : 3, for 19 : 2, for 5 : 0, for 13 : 0

Adding all these, we get : 2+5+1+1+3+2 = 14

So 14 swaps will be done.

#471 Explained Report Bookmark

An array of n numbers is given, where n is an even number. The maximum as well as the minimum of these n numbers needs to be determined. Which of the following is TRUE about the number of comparisons needed?

- A
  At least 2n-c comparisons, for some constant c, are needed
- B
  At most 1.5n-2 comparisons are needed.
- C
  At least nlog2n comparisons are needed.
- D
  None of the above.

Correct Answer :B

Explanation

Min: n-1 comparisions and Max: All those who lost in 1st round, they are all leaf nodes in tree... n/2 such persons are there.

So to find maximum from them....we need (n/2) - 1 comparisons.

Total: n-1 + (n/2)-1 = 1.5n -2

#472 Explained Report Bookmark

Which of the following abstract data types can be used to represent a many to many relation?

- A
  Tree
- B
  Plex
- C
  Graph
- D
  Both (b) and (c)

Correct Answer :D

Explanation

A plex (derived from plexus meaning any complex structure containing an intricate network of interrelated parts) consists of a set of elements called beads, where each element is an N-word vector of computer storage.

A Graph is a non-linear data structure consisting of nodes and edges. The nodes are sometimes also referred to as vertices and the edges are lines or arcs that connect any two nodes in the graph.

#473 Explained Report Bookmark

A binary tree in which if all its levels except possibly the last, have the maximum number of nodes and all the nodes at the last level appear as far left as possible, is called

- A
  Full binary tree
- B
  Binary Search Tree
- C
  Threaded tree
- D
  Complete binary tree

Correct Answer :D

Explanation

A binary tree in which if all its levels except possibly the last, have the maximum number of nodes and all the nodes at the last level appear as far as left as possible, is known as Computer binary tree.

#474 Explained Report Bookmark

The average successful search time taken by binary search on a sorted array of 10 items is

- A2.6
- B2.7
- C2.8
- D2.9

Correct Answer :D

Explanation

The 10 items i1, i2, ..., i10 may be arranged in a binary search trees as in Fig To get i5, the number of comparision needed is 1; for i2, it is 2; for i8 it is 2; for i1 it is 3, and so on. The average is (1+(2+2) +(3+3+3+3)+(4+4+4))/10, i.e., 2.9.

#475 Explained Report Bookmark

Which of the following remarks about Tree- indexing are true?

- A
  It is an m-ary tree
- B
  Successful searches should terminate in leaf nodes
- C
  Unsuccessful searches may terminate in leaf nodes level of the tree structure
- D
  All of these

Correct Answer :D

Explanation

B+Tree indexing is a method of accessing and maintaining data. It should be used for large files that have unusual, unknown, or changing distributions

m-ary tree (also known as k-ary or k-way tree) is a rooted tree in which each node has no more than m children. A binary tree is the special case where m = 2, and a ternary tree is another case with m = 3 that limits its children to three.

B-trees are always height balanced, with all leaf nodes at the same level

#476 Explained Report Bookmark

A complete binary tree of level 5 has how many nodes ?

- A
  15
- B
  25
- C
  63
- D
  30

Correct Answer :C

Explanation

A binary tree T with n levels is complete if all levels except possibly the last are completely full,and the last level has all its

nodes to the left side.

Let us consider a complete binary tree of height h (root at height 0)

1. minimum no of node =2^h
2. maximum no of node=2^{h+1}-1

3. No of leaf node=2^h
4. No of internal node=2^h-1

So in given question we need to find maximum no of node

h = 5

= $2^{5+1}-1$

=2^6-1

=64-1 =>63

#477 Explained Report Bookmark

What is the output of following function for start pointing to first node of following linked list? 1->2->3->4->5->6

```
void fun(struct node* start)

{

if(start == NULL)

return;

printf("%d ", start->data);



if(start->next != NULL )
```

fun(start->next->next);

printf("%d ", start->data);

}

- A1 4 6 6 4 1
- B1 3 5 1 3 5
- C1 2 3 5
- D1 3 5 5 3 1

Correct Answer :D

Explanation

fun() prints alternate nodes of the given Linked List, first from head to end, and then from end to head. If Linked List has even number of nodes, then skips the last node.

#478Explained Report Bookmark

A sorting technique that guarantees that records with the same primary key occurs in the same order in the sorted list as in the original unsorted list is said to be

- A
  stable
- B
  consistent
- C
  external
- D
  linear

Correct Answer :A

Explanation

Its called stablestable sorting technique.

This maintains the same position of the elements even after getting sorted.

Insertion, Bubble and Merge sorts are stable sorting algoritms.

#479Explained Report Bookmark

The following postfix expression with single digit operands is evaluated using a stack:

```
8 2 3 ^ / 2 3 * + 5 1 * -
```

Note that ^ is the exponentiation operator. The top two elements of the stack after the first * is evaluated are:

- A6, 1
- B5, 7
- C3, 2
- D1, 5

Correct Answer :A

Explanation

The algorithm for evaluating any postfix expression is fairly straightforward: 1. While there are input tokens left o Read the next token from input. o If the token is a value + Push it onto the stack. o Otherwise, the token is an operator (operator here includes both operators, and functions). * It is known a priori that the operator takes n arguments. * If there are fewer than n values on the stack (Error) The user has not input sufficient values in the expression. * Else, Pop the top n values from the stack. * Evaluate the operator, with the values as arguments. * Push the returned results, if any, back onto the stack. 2. If there is only one value in the stack o That value is the result of the calculation. 3. If there are more values in the stack o (Error) The user input has too many values. Source for algorithm: http://en.wikipedia.org/wiki/Reverse_Polish_notation#The_postfix_algorithm Let us run the above algorithm for the given expression. First three tokens are values, so they are simply pushed. After pushing 8, 2 and 3, the stack is as follows 8, 2, 3 When ^ is read, top two are popped and power(2^3) is calculated 8, 8 When / is read, top two are popped and division(8/8) is performed 1 Next two tokens are values, so they are simply pushed. After pushing 2 and 3, the stack is as follows 1, 2, 3 When * comes, top two are popped and multiplication is performed. 1, 6

#480 Explained Report Bookmark

Which one of the following statements is false?

- A
  Optimal binary search tree construction can be performed efficiently using dynamic programmmg.
- B
  Breadth-first search cannot be used to find connected components of a graph.
- C
  Given the prefix and postfix walks of a binary tree, the binary tree cannot be uniquely reconstructed.
- D
  Both (a) and (c)

Correct Answer :D

Explanation

(A) Optimal binary search tree construction can be performed efficiently using dynamic programming.

(B) Breadth-first search cannot be used to find connected components of a graph.

(C) Given the prefix and postfix walks of a binary tree, the tree cannot be re-constructed uniquely.

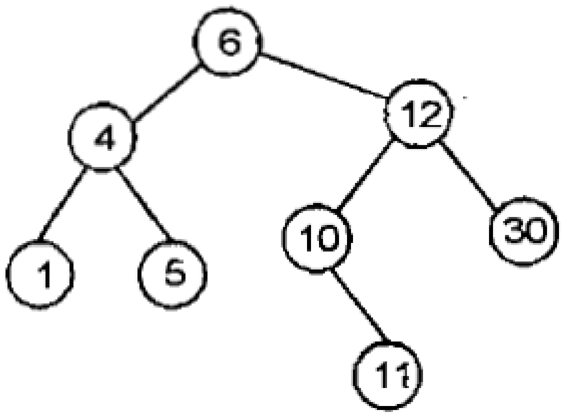
(D) Depth-first-search can be used to find the connected components of a graph.

All statements are correct except (B) because BFS can be used to check connectivity of graphs.

So, option (B) is correct.

#481 Explained Report Bookmark

If this tree is used for sorting, then new no 8 should be placed as the



- A
  left child of the node labelled 30
- B
  right child of the node labelled 5

- C
  right child of the node labelled 30
- D
  left child of the node labelled 10

Correct Answer :D

Explanation

As mentioned in the question, the tree is used for sorting. Hence performing inorder traversal (Left-Root-Right) we have 8 placed at the left child of the node labeled 10.

#482 Not Explained Report Bookmark

An algorithm is made up of 2 modules M1&M2. If order of M1 is f(n) & M2 is g(n) then the order of algorithm is?

- A max (f(n),g(n))
- B min (f(n),g(n))
- C f(n) + g(n)
- D f(n) X g(n)

Correct Answer :B

No Explanation Available

#483 Explained Report Bookmark

Traversing a binary tree first root and then left and right subtrees called _______traversal.

- A
  postorder
- B
  preorder

- C
  inorder
- D
  none of these

Correct Answer :B

Explanation

In a preorder traversal, each root node is visited before its left and right subtrees are traversed. Preorder search is also called backtracking.

The steps for traversing a binary tree in preorder traversal are:

1. Visit the root.
2. Visit the left subtree, using preorder.
3. Visit the right subtree, using preorder.

#484 Explained Report Bookmark

A full binary tree with n leaves contains

- A
  n nodes
- B
  log2n nodes
- C
  2n - 1 nodes
- D
  2n+1 nodes

Correct Answer :C

Explanation

If there are N leaves then there N/2 nodes at the level before and N/4 nodes at the level before. Thus the expression giving the total number of nodes is:

N(1/2^0 +1/2 +1/2^2+…1/2^ lg N) which equals N(2 - 1/2^(lg N)) = 2N - 1.

#485 Explained Report Bookmark

Sorting is useful for

- A
  report generation
- B
  responding toe queries easily
- C
  making searching easier and efficient
- D
  All of these

Correct Answer :D

Explanation

Sorting is useful for both report generation and making search easier and efficient . But sorting does not minimize the storage needed

#486 Explained Report Bookmark

Unrestricted use of goto is harmful, because it

- A
  makes debugging difficult.
- B
  increases the running time of programs

- C
  increases memory requirement of programs
- D
  results in the compiler generating longer machine code

Correct Answer :A

Explanation

Unrestricted use of goto statement is harmful because it makes more difficult to verify programs i.e. use of goto can results in unstructured code and there can be blocks with multiple entry and exit which can cause difficulty which debugging of program.

#487 Explained Report Bookmark

Is it possible to create a doubly linked list using only one pointer with every node. ?

- A
  Not Possible
- B
  Yes, possible by storing XOR of addresses of previous and next nodes.
- C
  Yes, possible by storing XOR of current node and next node
- D
  Yes, possible by storing XOR of current node and previous node

Correct Answer :B

Explanation

An ordinary Doubly Linked List requires space for two address fields to store the addresses of previous and next nodes. A memory efficient version of Doubly Linked List can be created using only one space for address field with every node. This memory efficient Doubly Linked List is called XOR Linked List or Memory Efficient as the list uses bitwise XOR operation to save space for one address. In the XOR linked list, instead of storing actual memory addresses, every node stores the XOR of addresses of previous and next nodes.

#488Explained Report Bookmark

L:et m, n be positive integers. Define Q (m, n) as

Q (m, n) = 0, if m>n

Then Q (m, 3) is (a div b, gives the quotient when a is divided by b)

- Aa constant
- Bp x (m mod 3)
- Cp x (m div 3)
- D3 x p

Correct Answer :C

Explanation

Let m>n. Let m/n yield a quotient x and remainder y. So, m= n*x+y and y

#489Explained Report Bookmark

You want to check whether a given set of items is sorted. Which of the following sorting methods will be most efficient if it is already in sorted order?

- A
  Bubble sort
- B
  Selection sort
- C
  Insertion sort
- D
  Merge Sort

Correct Answer :C

Explanation

- The standard Bubble sort implementation takes O(n2)O(n2) time in any case (though we have a modified version of it which can detect if there were any swaps made), but here we can't assume that the mentioned Bubble Sort procedure is modified.
- The Selection sort procedure is also going to take O(n2)O(n2) time.
- Merge Sort runs in Θ(nlogn)Θ(nlogn) time.
- The best algorithm to detect a sorted sequence is Insertion Sort. In case of sorted input, this procedure will terminate in O(n)O(n) time.

#490 Explained Report Bookmark

The way a card game player arranges his cards as he picks them up one by one, is an example of

- A
  bubble sort
- B
  Selection sort
- C
  insertion sort
- D
  merge sort

Correct Answer :C

Explanation

He scans through the rest of the cards and pick the one with the least value and places it next to the point till which he has already sorted the cards So insertion sort is Correct Answer

#491 Explained Report Bookmark

The minimum number of comparisons required to determine if an integer appears more than n/2 times in a sorted array of n integers is

- A
  Θ(n)
- B
  Θ(logn)
- C
  Θ(log*n)
- D
  Θ(1)

Correct Answer :B

Explanation

The Best way to find out whether an integer appears more than n/2 times in a sorted array(Ascending Order) of n integers, would be binary search approach.

1. The First occurence of an element can be found out in O(log(n)) time using divide and conquer technique,lets say it is i.
2. The Last occurrence of an element can be found out in O(log(n)) time using divide and conquer technique,lets say it is j.
3. Now number of occuerence of that element(count) is (j-i+1). Overall time complexity = log n +log n +1 = O(logn)

#492 Explained Report Bookmark

A linear collection of data elements where the linear node is given by means of pointer is called?

- A
  Linked list
- B
  Node list
- C
  Primitive list
- D
  None

Correct Answer :A

Explanation

In Linked list each node has its own data and the address of next node. These nodes are linked by using pointers. Node list is an object that consists of a list of all nodes in a document with in a particular selected set of nodes.

#493 Explained Report Bookmark

Consider an implementation of unsorted doubly linked list. Suppose it has its representation with a head pointer and tail pointer. Given the representation, which of the following operation can be implemented in O(1) time?

i) Insertion at the front of the linked list

ii) Insertion at the end of the linked list

iii) Deletion of the front node of the linked list

iv) Deletion of the end node of the linked list

- A
  I and II
- B
  I and III
- C
  I,II and III

- D
  I,II,III and IV

Correct Answer :C

Explanation

to delete the last node of the linked list, we need the address of the second last i.e. n-1th node of the single linked list to make NULL of its next pointer which will make second last node as a last node. But access to previous nodes in singly linked list is not possible,we must traverse entire linked list to get second last node of it.

#494 Explained Report Bookmark

Which of the following sorting methods will be the best if number of swappings done, is the only measure of efficienty?

- A Bubble sort
- B Selection sort
- C Insertion sort
- D Quick sort

Correct Answer :B

Explanation

Because in selection sort algorithm we randomly access data rather than a list in which we can easily swap data which we want to swap and it takes less time. So, answer is 'B'

#495 Explained Report Bookmark

You are asked to sort 15 randomly generated numbers. You should prefer

- A
  bubble sort
- B
  Selection sort
- C
  Insertion sort
- D
  Quick sort

Correct Answer :A

Explanation

Quite naturally one will not apply quick sort(first one have to understand it which itself is not easy)

insertion sort is best applicable when u can physically replace number (like in arranging cards)

selection sort is again not preferred for the same reason

bubble sort is best option as the   1)it is quite natural to adopt ,2) size of list is not large so swaps will not be issue and 3)it will give u partial sorted list in first few passes

#496 Explained Report Bookmark

As part of the maintenance work, you are entrusted with the work of rearranging the library books in a shelf in proper order, at the end of each day. The ideal choice will be

- A
  bubble sort
- B
  insertion sort
- C
  selection sort

- D
  heap sort

Correct Answer :B

Explanation

Rearranging the library books in a shelf in proper order is same as arranging cards. So insertion sort should be preferred.

#497 Explained Report Bookmark

The maximum number of comparisons needed to sort 7 items using radix sort is (assume each item is 4 digit decimal number)

- A 280
- B 40
- C 70
- D 38

Correct Answer :A

Explanation

The maximum number of comparison is number of items ´ radix ´ number of digits i.e., 7´10´4 = 280.

#498 Explained Report Bookmark

Which of the following algorithms exhibits the unnatural behavior that, minimum number of comparisons are needed if the list to be sorted is in the reverse sorted order and maximum number of comparisons are needed if they are already in sorted order?

- A
  heap sort

- B
  Radix sort
- C
  Binary insertion sort
- D
  There can't be any such sorting method

Correct Answer :C

Explanation

Binary insertion sort take minimum comparison if the list to be sorted is in reverse order and maximum number of comparison if they already in sorted order.

#499 Explained Report Bookmark

Which of the following sorting methods sorts a given set of items that is already in sorted order or in reverse sorted order with equal speed?

- A
  Heap sort
- B
  Quick sort
- C
  Insertion sort
- D
  Selection sort

Correct Answer :B

Explanation

Quick sort has two worst cases, when input is in either ascending or descending order, it takes same time O(n2).

#500 Explained Report Bookmark

Which of the following algorithm design technique is used in the quick sort algorithm?

- A
  Dynamic programming
- B
  Backtracking
- C
  Divide and conquer
- D
  Greedy method

Correct Answer :C

Explanation

A Divide and Conquer Algorithm breaks up a problem into many parts ("Divide") and then solves each part individually ("Conquer")

Quick Sort is a sorting Algorithm.

In Quick Sort, to sort an array:

-> We first chose a "pivot" element of the array. This divides the array in two parts: One part of elements before the pivot, and one part is after that.

-> Now, we reorder the array so that all elements smaller than the pivot come before the pivot, and all elements larger than the pivot come after the pivot.

-> The above steps are recursively applied to sub-arrays, one containing smaller values, and one containing larger values

Essentially, Quick Sort divides an Array into Sub Arrays, and works on each part individually.

So, Quick Sort is a Divide and Conquer Algorithm.

#501 Explained Report Bookmark

The maximum number of nodes on level 5 of a binary tree.

null

- A
  16
- B
  32
- C
  48
- D
  63

Correct Answer :A

Explanation

The maximum number of nodes on level 5 of a binary is 16. The maximum number of nodes in a binary tree of depth k is 2^k− 1, k>=1.

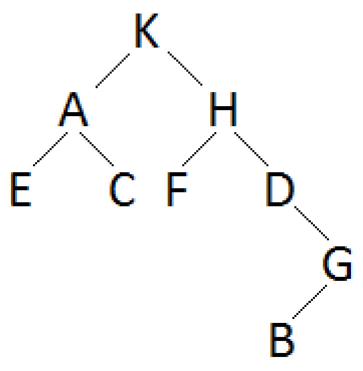
#502 Explained Report Bookmark

When inorder traversing a tree resulted E A C K F H D B G; the preorder traversal would return

- A
  FAEKCDBHG
- B
  FAEKCDHGB
- C
  EAFKHDCBG
- D
  FEAKDCHBG

Correct Answer :B

Explanation



Inorder: E A C K F H D B G

Preorder: K A E C H F D G B

#503 Explained Report Bookmark

The worst case occur in quick sort when

- A
  I.Pivot is the middle most element of an array and this happens when input array is sorted or reverse sorted
- B
  II. Pivot is the left most element and this happens when input array is sorted or reverse sorted
- C
  III. Pivot is the right most element and This happens when input array is sorted or reverse sorted
- D
  II & III

Correct Answer :D

Explanation

The efficiency of the Quicksort algorithm very much depends on the selection of the pivot element.

Ascending Sorted+left: Let’s assume the input of the Quicksort is a sorted array and we choose the leftmost element as a pivot element. In this case, we’ll have two extremely unbalanced arrays. One array will have one element and the other one will have (N - 1) elements.

Reversely sorted+right: Similarly, when the given input array is sorted reversely and we choose the rightmost element as the pivot element, the worst case occurs. Again, in this case, the pivot elements will split the input array into two unbalanced arrays

#504 Explained Report Bookmark

In the following type of searching key-comparisons are needed

- A
  I. Linear Search
- B
  II. Non Linear Search
- C
  Both I and II
- D
  None of these

Correct Answer :C

Explanation

Linear Search: It’s used to find an element in an array by looping over the elements of the array, and in each iteration, we compare each element with the required element, and when the required element is found, its index is returned, and in case of no match is found we can return -1.

Non-Linear Search / Binary Search: This search algorithm is used with sorted array to find a specific element and return its index if found, or return -1 if not found.

#505 Explained Report Bookmark

Which of the following algorithms has lowest worst case time complexity

null

- A
  Insertion Sort

- B
  Selection Sort
- C
  Quick Sort
- D
  merge Sort

Correct Answer :D

Explanation

Worst case complexity of merge sort is O(nlogn).

#506 Explained Report Bookmark

Binary search algorithm cannot be applied to

null

- A
  sorted linear array
- B
  sorted binary trees
- C
  sorted linked list
- D
  All of the above

Correct Answer :D

Explanation

Binary Search Algorithm can be applied only on Sorted arrays.

#507 Explained Report Bookmark

Items in a priority queue are entered in a _______________ order

null

- A
  random
- B
  order of priority
- C
  as and when they come
- D
  none of the above

Correct Answer :A

Explanation

Items in a priority queue are entered in a random order. Priority queue contains data items which have some preset priority. While removing an element from a priority queue, the data item with the highest priority is removed first.

#508 Explained Report Bookmark

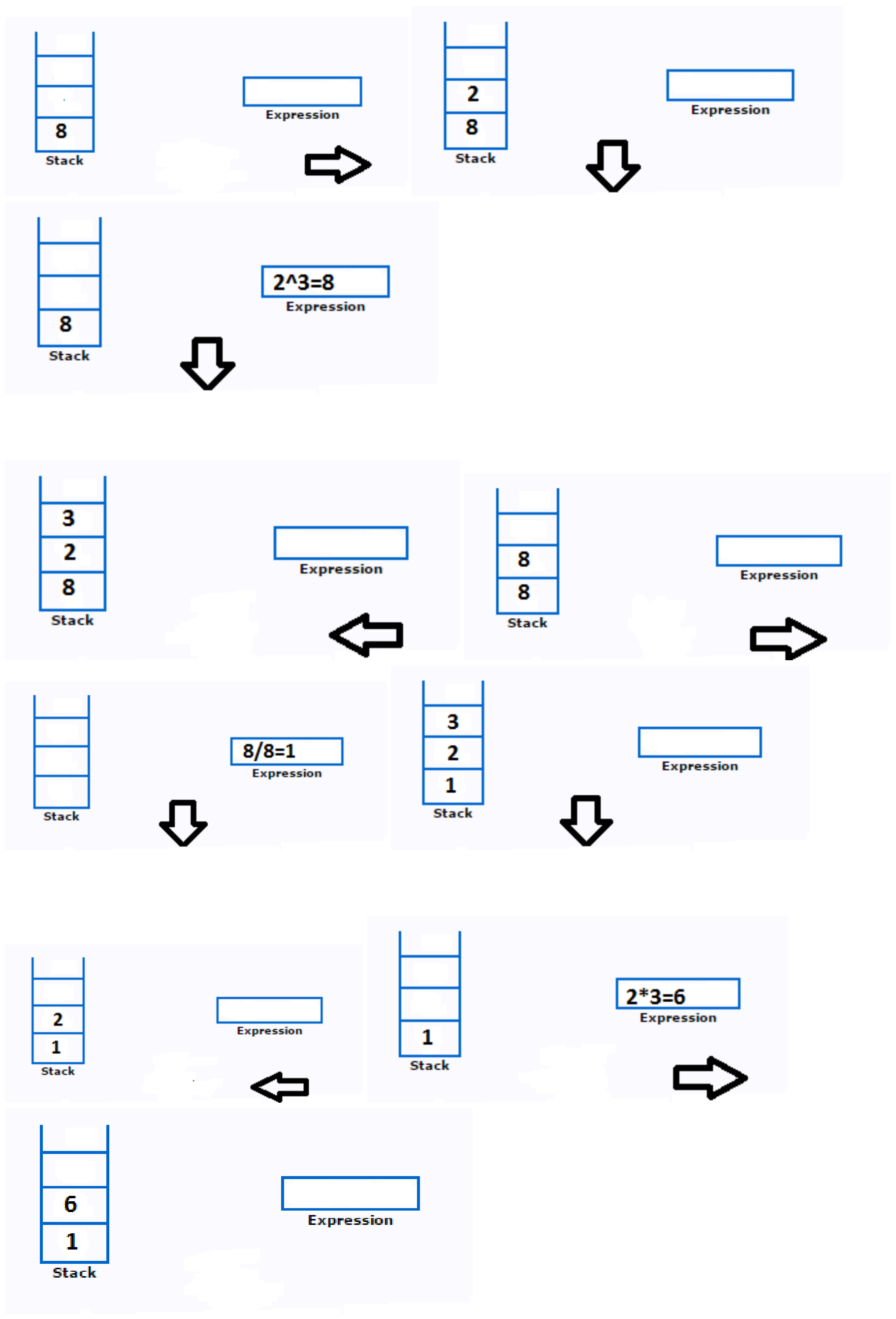
The following postfix expression is evaluated using a stack 823^/23* + 51* – The top two elements of the stack after first * is evaluated

null

- A
  6, 1
- B
  5,7
- C
  3.2
- D
  1,5

Correct Answer :A

Explanation



Hence, Option(A)6,1 is the correct choice.

Stack : 8 2 3 ^

Stack : 8 8 /

Stack : 1 2 3 *

Stack : 1 6

#509 Explained Report Bookmark

Consider an implementation of unsorted single linked list. Suppose it has its representation with a head and a tail pointer (i.e. pointers to the first and last nodes of the linked list). Given the representation, which of the following operation can not be implemented in O(1) time ?

null

- A
  Insertion at the front of the linked list.
- B
  Insertion at the end of the linked list.
- C
  Deletion of the front node of the linked list.
- D
  Deletion of the last node of the linked list.

Correct Answer :D

Explanation

Deletion of the last node of the linked list, we need address of second last node of single linked list to make NULL of its next pointer. Since we can not access its previous node in singly linked list, so need to traverse entire linked list to get second last node of linked list.

#510 Explained Report Bookmark

The depth of a complete binary tree with ‘n’ nodes is (log is to the base two)

null

- A
  log(n+1)-1
- B
  log(n-1) + 1
- C
  log(n)
- D
  log(n) + 1

Correct Answer :A

Explanation

Total no. of nodes n = (2^(d+1))-1

log (n+1) = d+1

d = log(n+1)-1

#511 Explained Report Bookmark

If the sequence of operations push(1), push(2), pop,push (1),push(2), pop, pop, pop, push (2), pop are performed on a stack, the sequence of popped out values are

null

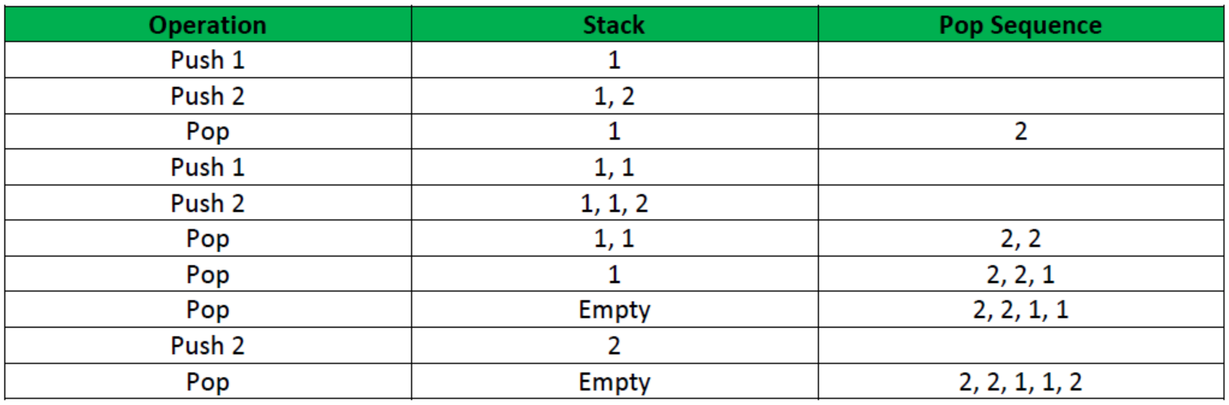
- A
  2, 2, 1, 1, 2
- B
  2, 2, 1, 2, 2
- C
  2,1, 2, 2, 2

- D
  2, 1, 2, 2, 1

Correct Answer :A

Explanation

The pop sequence can be seen from the following table:



Operation	Stack	Pop Sequence
Push 1	1	
Push 2	1, 2	
Pop	1	2
Push 1	1, 1	
Push 2	1, 1, 2	
Pop	1, 1	2, 2
Pop	1	2, 2, 1
Pop	Empty	2, 2, 1, 1
Push 2	2	
Pop	Empty	2, 2, 1, 1, 2

#512Explained Report Bookmark

Given two sorted list of size ‘m and ‘n’ respectively. The number of comparisons needed in the worst case by the merge sort algorithm will be

null

- A
  m * n
- B
  minimum of m, n
- C
  maximum of m, n
- D
  m+n-1

Correct Answer :D

Explanation

The number of comparisons needed in the worst case by the merge sort algorithm will be m+n-1 i.e. only last element will not be compare only.

#513 Explained Report Bookmark

In linked lists there are no NULL links in

- A
  single linked list
- B
  linear doubly linked list
- C
  circular linked list
- D
  linked list

Correct Answer :C

Explanation

A circular linked list is a sequence of elements in which every element has a link to its next element in the sequence and the last element has a link to the first element.

#514 Explained Report Bookmark

Each node in the singly linked list has …….. fields.

null

- A
  2
- B
  3

- C
  1
- D
  4

Correct Answer :A

Explanation

a linked list consists of 2 nodes where each node contains a data field and a reference(link) to the next node in the list.

#515 Explained Report Bookmark

Which of the following is two way lists?

null

- A
  Grounded header list
- B
  Circular header list
- C
  Linked list with header and trailer nodes
- D
  List traversed in two directions

Correct Answer :D

Explanation

for doubly linked list is an example of linked in list that can be traversed from both sides.

#516 Explained Report Bookmark

Which is the pointer associated with the availability list?

null

- A
  FIRST
- B
  AVAIL
- C
  TOP
- D
  REAR

Correct Answer :B

Explanation

It is a list of free nodes in the memory. It contains unused memory cells and these memory cells can be used in future. It is also known as ‘List of Available Space’ or ‘Free Storage List’ or ‘Free Pool’. It is used along with linked list. Whenever a new node is to be inserted in the linked list, a free node is taken from the AVAIL List and is inserted in the linked list. Similarly, whenever a node is deleted from the linked list, it is inserted in the AVAIL List. So that it can be used in future..

#517 Explained Report Bookmark

Value of first linked list index is ….

null

- A
  0
- B
  1
- C
  -1

- D
  2

Correct Answer :A

Explanation

in linked list, valid Index starts from 0 and ends on (N-1) , where N=size of the linked list.

#518 Explained Report Bookmark

Each node in a linked list must contain at least .....

null

- A
   Three fields
- B
  Two fields
- C
  Four fields
- D
  Five fields

Correct Answer :B

Explanation

In a simple linked list each link of a linked list can store a data called an element. Each node contains two fields, called links that are references to the previous and to the next node in the sequence of nodes.

#519 Explained Report Bookmark

The dummy header in linked list contain .....

null

- A
  first record of the actual data
- B
  last record of the actual data
- C
  pointer to the last record of the actual data
- D
  middle record of the actual data

Correct Answer :A

Explanation

they allow you to write general code that works for empty and non-empty lists.if you use a dummy header, head will never be null. It will be a Node with a null next pointer, which you can point to your new Node

#520Explained Report Bookmark

In a linked list the ………. field contains the address of next element in the list.

null

- A
  Link field
- B
  Next element field
- C
  Start field
- D
  Info field

Correct Answer :A

Explanation

Each record of a linked list is often called an 'element' or 'node'. The field of each node that contains the address of the next node is usually called the 'next link' or 'next pointer'.

#521 Explained Report Bookmark

LLINK is the pointer pointing to the

null

- A
  successor node
- B
  predecessor node
- C
  head node
- D
  last node

Correct Answer :B

Explanation

a node may also have a reference to the node containing the previous value in the sequence. This is the predecessor reference.

#522 Explained Report Bookmark

………. refers to a linear collection of data items.

null

- A
  List
- B
  Tree
- C
  Graph

- D
  Edge

Correct Answer :A

Explanation

A linear collection of data elements is called List. a list or sequence is an abstract data type that represents a countable number of ordered values, where the same value may occur more than once.

#523 Explained Report Bookmark

A run list is ……

null

- A
  small batches of records from a file
- B
  number of elements having same value
- C
  number of records
- D
  number of files in external storage

Correct Answer :A

Explanation

A Run List is a collection of one or more automation flows that are executed on one or more Schedules

#524 Explained Report Bookmark

A …… indicates the end of the list.

null

- A
  Guard
- B
  Sentinel
- C
  End pointer
- D
  Last pointer

Correct Answer :B

Explanation

In computer programming, a sentinel node is a specifically designated node used with linked lists and trees as a traversal path terminator. This type of node does not hold or reference any data managed by the data structure.

#525 Explained Report Bookmark

A …….. is a linear list in which insertions and deletions are made to from either end of the structure.

null

- A
  circular queue
- B
  random of queue
- C
  priority
- D
  dequeue

Correct Answer :D

Explanation

Dequeue means double ended queue.It can act as queue from front as well as from rear side.Thats why both insertion and deletion is possible from either end.

#526 Explained Report Bookmark

Indexing the …….. element in the list is not possible in linked lists.

null

- A
  middle
- B
  first
- C
  last
- D
  any where in between

Correct Answer :A

Explanation

In order to locate the middle element in linked list one must traverse whole list first and then again traverse the same list to locate the middle element based on first traverse result i.e. length.Thus it is impossible to traverse whole list at once since it can contain n number of elements and will affect performance.

#527 Explained Report Bookmark

A linear list in which the pointer points only to the successive node is ……

null

- A
  singly linked list

- B
  circular linked list
- C
  doubly linked list
- D
  none of the above

Correct Answer :A

Explanation

A node contains two fields i.e. data stored at that particular address and the pointer which contains the address of the next node in the memory.

#528 Explained Report Bookmark

………. may take place only when there is some minimum amount(or) no space left in free storage list.

null

- A
  Memory management
- B
  Garbage collection
- C
  Recycle bin
- D
  Memory management

Correct Answer :B

Explanation

It is a technique in which the operating system periodically collects all the deleted space onto the free storage list. It takes place when there is minimum amount of space left in storage list or when CPU is ideal.

#529 Explained Report Bookmark

A linear list in which the last node points to the first node is ……..

null

- A
  singly linked list
- B
  circular linked list
- C
  doubly linked list
- D
  none of the above

Correct Answer :B

Explanation

A circular linked list is a sequence of elements in which every element has a link to its next element in the sequence and the last element has a link to the first element. That means circular linked list is similar to the single linked list except that the last node points to the first node in the list.

#530 Explained Report Bookmark

a new node in linked list free node will be available in ……..

null

- A
  Available list
- B
   Avail list

- C
  Free node list
- D
  Memory space list

Correct Answer :B

Explanation

It is a list of free nodes in the memory. It contains unused memory cells and these memory cells can be used in future. It is also known as 'List of Available Space' or 'Free Storage List' or 'Free Pool'. It is used along with linked list. Whenever a new node is to be inserted in the linked list, a free node is taken from the AVAIL List and is inserted in the linked list. Similarly, whenever a node is deleted from the linked list, it is inserted in the AVAIL List. So that it can be used in future.

#531 Explained Report Bookmark

A singly linked list is also called as ……..

null

- A
  linked list
- B
  one way chain
- C
  two way chain
- D
  right link

Correct Answer :B

Explanation

A singly linked list is a type of linked list that is unidirectional, that is, it can be traversed in only one direction from head to the last node (tail).

#532 Explained Report Bookmark

A ….. list is a header list where the node points back to the header node.

null

- A
  Circular header
- B
  Grounded header
- C
  Two way header
- D
  One way header

Correct Answer :A

Explanation

A list in which last node points back to the header node is called circular linked list. The chains do not indicate first or last nodes. In this case, external pointers provide a frame of reference because last node of a circular linked list does not contain the NULL pointer.

#533 Explained Report Bookmark

A doubly linked list has ………. pointers with each node.

null

- A
  0
- B
  1
- C
  2
- D
  3

Correct Answer :C

Explanation

Doubly linked list is a type of linked list in which each node apart from storing its data has two links. The first link points to the previous node in the list and the second link points to the next node in the list.

#534 Explained Report Bookmark

Header linked lists are frequently used for maintaining …….. in memory.

null

- A
  Polynomials
- B
  Binomial

- C
  Trinomial
- D
  Quadratic equation

Correct Answer :A

Explanation

he header linked lists are frequently used to maintain the polynomials in memory. The header node is used to represent the zero polynomial. Suppose we have F(x) = 5x5 – 3x3 + 2x2 + x1 +10x0 From the polynomial represented by F(x) it is clear that this polynomial has two parts, coefficient and exponent, where, x is formal parameter. Hence, we can say that a polynomial is sum of terms, each of which consists of a coefficient and an exponent.

#535 Explained Report Bookmark

The pointer that points to the first node in the list is ……..

null

- A
  FIRST
- B
  AVAIL
- C
  TOP
- D
  REAR

Correct Answer :A

Explanation

A linked list is represented by a pointer to the first node of the linked list. The first node is called the head.

#536 Explained Report Bookmark

Two-way list may be maintained in memory by means of …………

null

- A
  Queues
- B
  Linear arrays
- C
  Non linear arrays
- D
  Stacks

Correct Answer :B

Explanation

A linear array, is a list of finite numbers of elements stored in the memory. In a linear array, we can store only homogeneous data elements. Elements of the array form a sequence or linear list, that can have the same type of data. Each element of the array, is referred by an index set.

#537 Explained Report Bookmark

A doubly linked list is also called as ……….

null

- A
  linked list
- B
  one way chain
- C
  two way chain
- D
  right link

Correct Answer :C

Explanation

Each node contains three fields: two link fields (references to the previous and to the next node in the sequence of nodes) and one data field.

#538 Explained Report Bookmark

The list that requires two pointer variables FIRST and LAST is called ……..

null

- A
  Circular list
- B
  Header list
- C
  One way list
- D
  Two way list

Correct Answer :D

Explanation

A two way list is a linear collection of data elements, called nodes, where each node N is divided into three parts:- information field, Forward link- which points to the next node and Backward link-which points to the previous node.

#539 Explained Report Bookmark

If the availability list is null, then the condition is said to be ………

null

- A
  nil block
- B
  availability list underflow
- C
  availability list overflow
- D
  memory loss

Correct Answer :B

Explanation

Underflow happens at the time of deletion. If we have to delete data from the data structure, but there is no data in the data structure i.e. data structure is empty, then this situation is called 'Underflow'.

#540 Explained Report Bookmark

The list which has its own pointer is called ……..

null

- A
  pointer list
- B
  self pointer

- C
  free pool
- D
  own pointer

Correct Answer :C

Explanation

Pool is a list consisting of unused memory cells which has its own pointer

#541 Explained Report Bookmark

Which of the following is two way lists?

null

- A
  Grounded header list
- B
  Circular header list
- C
  Linked list with header and trailer nodes
- D
  None of the above

Correct Answer :D

Explanation

List traversed in two directions is the answer

#542 Explained Report Bookmark

A .... is a header list where the last node contains the null pointer.

null

- A
  grounded header list
- B
  bottom header list
- C
  down header list
- D
   dropped header list

Correct Answer :A

Explanation

It is a list whose last node contains the NULL pointer. In the header linked list the start pointer always points to the header node. start -> next = NULL indicates that the grounded header linked list is empty.

#543 Explained Report Bookmark

RLINK is the pointer pointing to the ...

null

- A
  successor node
- B
   predecessor node
- C
   head node
- D
  last node

Correct Answer :A

Explanation

successor node is the second link that points to the next node in the list.

#544 Explained Report Bookmark

A …………. is a header list where the last node points back to the header node.

null

- A
  rounded header list
- B
  circular header list
- C
  common header list
- D
  forward header list

Correct Answer :B

Explanation

A list in which last node points back to the header node is called circular linked list. The chains do not indicate first or last nodes. In this case, external pointers provide a frame of reference because last node of a circular linked list does not contain the NULL pointer

#545 Explained Report Bookmark

In a linked list, insertion can be done as ………

null

- A
  beginning
- B
  end
- C
  middle

- D
  all of the above

Correct Answer :D

Explanation

Inserting a new element in an array of elements is expensive because the room has to be created for the new elements and to create room existing elements have to be shifted. But since linked list store elements in random memory locations using pointer instead of in sequential order, it is easy to add an element anywhere in the list and change the required pointers.

#546 Explained Report Bookmark

In a two-way lists each node is divided into …….parts.

null

- A
  1
- B
  2
- C
  3
- D
  4

Correct Answer :C

Explanation

A two-way list is a linear collection of data elements, called nodes, where each node N is divided into three parts: – 1. Information field – 2. Forward Link which points to the next node – 3. Backward Link which points to the previous node

#547 Explained Report Bookmark

The disadvantage in using a circular linked list is …….

null

- **A**
  it is possible to get into infinite loop
- **B**
  last node points to fist node.
- **C**
  time consuming
- **D**
  requires more memory space.

Correct Answer :A

Explanation

A circular linked list is a sequence of elements in which every element has a link to its next element in the sequence and the last element has a link to the first element. This means the linked list can be traversed n number of times.

#548 Explained Report Bookmark

Which of the following conditions checks available free space in avail list?

null

- **A**
  Avail=Null
- **B**
  . Null=Avail
- **C**
  Avail=Max stack
- **D**
  Avail=Top

Correct Answer :A

Explanation

Overflow is a condition that occurs when AVAIL = NULL or no free memory cell is present in the system. When this condition occurs, the program must give an appropriate message

#549 Explained Report Bookmark

A linear list in which each node has point to the predecessor and successors nodes is called ……..

null

- A
  singly linked list
- B
  circular linked list
- C
  doubly linked list
- D
  linear linked list

Correct Answer :C

Explanation

Doubly linked list is a type of linked list in which each node apart from storing its data has two links. The first link points to the previous node in the list and the second link points to the next node in the list.

#550 Explained Report Bookmark

The queue in which the insertion takes place in the first position after of last element is a ……

null

- A
  priority
- B
  dequeue
- C
  circular
- D
  linked

Correct Answer :C

Explanation

A circular linked list is a sequence of elements in which every element has a link to its next element in the sequence and the last element has a link to the first element.

#551 Explained Report Bookmark

Before inserting into stack one must check the condition ....

null

- A
  Overflow
- B
  Underflow
- C
  Maximum elements
- D
  Existing elements

Correct Answer :A

Explanation

A stack overflow is an undesirable condition in which a particular computer program tries to use more memory space than the call stack has available

#552Explained Report Bookmark

The another name of deque is ………

- A
  divided queue
- B
  distributed queue
- C
  double ended queue
- D
  design queue

Correct Answer :C

Explanation

A deque, also known as a double-ended queue, is an ordered collection of items similar to the queue.

#553Explained Report Bookmark

Before deletion condition into stack …… has to be checked

null

- A
  Overflow
- B
  Underflow
- C
  Maximum elements
- D
  Existing elements

Correct Answer :B

Explanation

When a stack is empty (i.e. TOP= -1) and we try to delete more element from it, then this condition is called underflow condition.

#554 Explained Report Bookmark

In dequeue, insertion and deletion takes place of ……….

null

- A
  front, rear end
- B
  only at rear end
- C
  only at front end
- D
  both the ends

Correct Answer :D

Explanation

A deque, also known as a double-ended queue, is an ordered collection of items similar to the queue. It has two ends, a front and a rear, and the items remain positioned in the collection.

#555 Explained Report Bookmark

When does Top value of stack change in insertion process?

null

- A
  Before insertion
- B
  After insertion
- C
  At the time of insertion
- D
  While checking overflow

Correct Answer :A

Explanation

In order to insert new element top value must be first increased and then insertion takes place

#556 Explained Report Bookmark

A queue in which insertion and deletion takes places from any position is called ……

null

- A
  circular queue
- B
  random of queue
- C
  priority
- D
  dequeue

Correct Answer :C

Explanation

a priority queue is an abstract data type similar to a regular queue or stack data structure in which each element additionally has a "priority" associated with it. In a priority queue, an element with high priority is served before an element with low priori

#557 Explained Report Bookmark

Deletion in the linked stack takes place by deleting ……..

null

- A
  Node pointed by the start process.
- B
  End of the list
- C
  Beginning of the list
- D
  Middle of the list

Correct Answer :A

Explanation

Deleting a node from the top of stack is referred to as pop operation. Deleting a node from the linked list implementation of stack is different from that in the array implementation.

#558 Explained Report Bookmark

Which of the following name does not relate to stacks?

null

- A
  FIFO lists
- B
  LIFO list

- C
  piles
- D
  push-down lists

Correct Answer :A

Explanation

stack is a LIFO i.e. Last In First Out.

#559 Explained Report Bookmark

The condition …….. indicate the queue is empty.

null

- A
   Front=Null
- B
  Null=Front
- C
  Front=Rear
- D
   Rear=Null

Correct Answer :A

Explanation

 Front indicates the first node of the queue. So technically if front is null then there is no node present.

#560 Explained Report Bookmark

Herder node is used as sentinel in …..

null

- A
  Graphs
- B
  Stacks
- C
  Binary tree
- D
  Queues

Correct Answer :C

Explanation

The header has a valid next reference by a null prev reference, while the trailer has a valid prev reference by a null next reference. A linked list object would simply need to store references to these two sentinels and a size counter that keeps track of the number of elements (not counting sentinels) in the list.

#561 Explained Report Bookmark

The value of REAR is increased by 1 when …….

null

- A
  An element is deleted in a queue
- B
  An element is traversed in a queue
- C
  An element is added in a queue
- D
  An element is merged in a queue

Correct Answer :C

Explanation

REAR is the last node of the queue. Thus in order to add new element REAR must be increased and then added to the queue.

#562 Explained Report Bookmark

The operations that can be done in a circular queue is/are …..

null

- A
  Insert from the front end
- B
  Delete from front end
- C
  Display queue contents
- D
  All of the above

Correct Answer :D

Explanation

Insert from the front end, Delete from front end, Display queue contents all three operations are possible with circular queue.

#563 Explained Report Bookmark

The term dequeue is the contraction of the name ……..

null

- A
  Double ended queu
- B
  Double sided queue

- C
  Double headed queue
- D
  Double address queue

Correct Answer :A

Explanation

DEQUEUE is also called as double ended queue. It is an ordered collection of items similar to the queue. It has two ends, a front and a rear, and the items remain positioned in the collection.

#564 Explained Report Bookmark

The various operations that can be performed on stacks is/are .....

null

- A
  Insert an item into the stack
- B
  Delete an item from the stack
- C
  Display the contents of the stack
- D
  All of the above

Correct Answer :D

Explanation

Insert an item into the stack, Delete an item from the stack, Display the contents of the stack are all possible with stack.

#565 Explained Report Bookmark

………... is a collection of elements such that each element has been assigned a processing priority.

null

- A
  Priority queue
- B
  Procedure queue
- C
  Main queue
- D
  Interrupt queue

Correct Answer :A

Explanation

In computer science, a priority queue is an abstract data type similar to a regular queue or stack data structure in which each element additionally has a "priority" associated with it. In a priority queue, an element with high priority is served before an element with low priority.

#566 Explained Report Bookmark

Which data structure is used in breadth first search of a graph to hold nodes?

null

- A
  Stack
- B
  queue
- C
  Tree
- D
  Array

Correct Answer :B

Explanation

Breadth First Search (BFS) algorithm traverses a graph in a breadthward motion and uses a queue to remember to get the next vertex to start a search, when a dead end occurs in any iteration.

#567 Explained Report Bookmark

Link fields holds pointers to the ………. element in the linked representation of stack.

null

- A
  Neighboring
- B
  Last
- C
  First
- D
  Middle

Correct Answer :A

Explanation

The pointer which gives the location of the next node in the list is one of the field of stack node.

#568 Explained Report Bookmark

The pointer associated with the stack is ………..

null

- A
  front
- B
  rear
- C
  top
- D
  link

Correct Answer :C

Explanation

As this pointer always represents the top of the stack, hence named top. The top pointer provides top value of the stack without actually removing it.

#569 Explained Report Bookmark

Reversing a great deal of space for each stack in memory will ………..

- A
  Decrease the numbers of times overflow may occur
- B
  Increase the numbers of times overflow may occur
- C
  Increase the number of times underflow may occur
- D
  Decrease the number of times underflow may occur

Correct Answer :A

Explanation

A stack overflow is an undesirable condition in which a particular computer program tries to use more memory space than the call stack has available. Thus space will directly affect the overflow count.

#570 Explained Report Bookmark

Which of the following data structure is non linear type?

null

- A
  Strings
- B
  Lists
- C
  Stacks
- D
  Graph

Correct Answer :D

Explanation

The Data structure is said to be Non-Linear Data Structures if it's elements do not form a sequence or a linear series but form a hierarchical format. For Example:- Graph, Tree, Binary Search Tree etc.

#571 Explained Report Bookmark

Which of the following data structure is linear type?

null

- A
  Graph
- B
  Trees
- C
  Binary tree
- D
  Stack

Correct Answer :D

Explanation

Linear data structure: A linear data structure traverses the data elements sequentially, in which only one data element can directly be reached. Ex: Arrays, Linked Lists, Stack, Queue, Any type of List all are linear.

#572 Explained Report Bookmark

Finding the location of a given item in a collection of items is called …

null

- A
  Discovering
- B
  Finding
- C
  Searching
- D
  Mining

Correct Answer :C

Explanation

Searching is an operation or a technique that helps finds the place of a given element or value in the list. Any search is said to be successful or unsuccessful depending upon whether the element that is being searched is found or not.

#573 Explained Report Bookmark

Which of the following is an external sorting?

null

- A
  Insertion Sort
- B
  Bubble Sort
- C
  Merge Sort
- D
  Tree Sort

Correct Answer :C

Explanation

External sorting is a class of sorting algorithms that can handle massive amounts of data. External sorting is required when the data being sorted do not fit into the main memory of a computing device (usually RAM) and instead they must reside in the slower external memory, usually a hard disk drive.

#574 Explained Report Bookmark

Very slow way of sorting is ……….

null

- A
  Insertion sort
- B
  Heap sort
- C
  Bubble sort
- D
  Quick sort

Correct Answer :A

Explanation

Worst case is reverse sorted array, every comparison the inner loop will go over the whole sub-array.

#575Explained Report Bookmark

Which of the following is an internal sorting?

null

- A
  Tape Sort
- B
  2-way Merge Sort
- C
  Merge Sort
- D
  Tree Sort

Correct Answer :D

Explanation

An internal sort is any data sorting process that takes place entirely within the main memory of a computer. This is possible whenever the data to be sorted is small enough to all be held in the main memory.

#576Explained Report Bookmark

Sorting a file F usually refers to sorting F with respect to a particular key called …..

null

- A
  Basic key
- B
  Primary key
- C
  Starting key

- D
  Index key

Correct Answer :B

Explanation

The primary key's main function is to uniquely identify each row in the table - but it doesn't imply any (physical) sorting per sec.

#577 Explained Report Bookmark

The time complexity of quick sort is ……..

null

- A
  O(n)
- B
  O(logn)
- C
  O(n2)
- D
   O(n logn)

Correct Answer :D

Explanation

To sort an array of n distinct elements, quicksort takes O(n log n) time in expectation, averaged over all n! permutations of n elements with equal probability.

#578 Explained Report Bookmark

Selection sort first finds the ………. element in the list and put it in the first position.

null

- A
  Middle element
- B
  Largest element
- C
  Last element
- D
  Smallest element

Correct Answer :D

Explanation

This sorting algorithm is an in-place comparison-based algorithm in which the list is divided into two parts, the sorted part at the left end and the unsorted part at the right end. Initially, the sorted part is empty and the unsorted part is the entire list.

#579 Explained Report Bookmark

Quick sort is also known as ……..

null

- A
  merge sort
- B
  tree sort
- C
  shell sort
- D
  partition and exchange sort

Correct Answer :D

Explanation

The key process in quickSort is partition(). Target of partitions is, given an array and an element x of array as pivot, put x at its correct position in sorted array and put all smaller elements (smaller than x) before x, and put all greater elements (greater than x) after x.

#580 Explained Report Bookmark

The operation that combines the element is of A and B in a single sorted list C with n=r+s element is called ….

null

- A
  Inserting
- B
  Mixing
- C
  Merging
- D
  Sharing

Correct Answer :C

Explanation

Merge algorithms are a family of algorithms that take multiple sorted lists as input and produce a single list as output, containing all the elements of the inputs lists in sorted order.

#581 Explained Report Bookmark

A tree sort is also known as ……… sort.

null

- A
  quick
- B
  shell
- C
  heap
- D
  selection

Correct Answer :C

Explanation

Heap sort is a comparison based sorting technique based on Binary Heap data structure. It is similar to selection sort where we first find the maximum element and place the maximum element at the end. We repeat the same process for the remaining elements. Thus it is also called as heapsort.

#582 Explained Report Bookmark

………. sorting is good to use when alphabetizing large list of names.

null

- A
  Merge
- B
  Heap
- C
  Radix
- D
  Bubble

Correct Answer :C

Explanation

Radix sort is a sorting technique that sorts the elements by first grouping the individual digits of the same place value. Then, sort the elements according to their increasing/decreasing order. As compared to other sorting methods radix sort handles alphanumeric lists in a well manner.

#583 Explained Report Bookmark

The easiest sorting is ……..

null

- A
  quick sort
- B
  shell sort
- C
  heap sort
- D
  . selection sort

Correct Answer :D

Explanation

Selection sort algorithm is an in-place comparison-based algorithm in which the list is divided into two parts, the sorted part at the left end and the unsorted part at the right end. Initially, the sorted part is empty and the unsorted part is the entire list. The smallest element is selected from the unsorted array and swapped with the leftmost element, and that element becomes a part of the sorted array. This process continues moving unsorted array boundary by one element to the right.

#584 Explained Report Bookmark

Which of the following sorting algorithm is of divide and conquer type?

null

- A
  Bubble sort
- B
  Insertion sort
- C
  Quick sort
- D
  Merge sort

Correct Answer :C

Explanation

. A divide-and-conquer algorithm recursively breaks down a problem into two or more sub-problems of the same or related type, until these become simple enough to be solved directly. This same method is used by quick sort too.

#585 Explained Report Bookmark

Merging k sorted tables into a single sorted table is called ……

null

- A
  k way merging
- B
  k th merge
- C
  k+1 merge
- D
  k-1 merge

Correct Answer :A

Explanation

k-way merge algorithms or multiway merges are a specific type of sequence merge algorithms that specialize in taking in k sorted lists and merging them into a single sorted list.

#586 Explained Report Bookmark

The function used to modify the way of sorting the keys of records is called ……..

null

- A
  Indexing function
- B
  Hash function
- C
  Addressing function
- D
  All of the above

Correct Answer :B

Explanation

A hash function takes an input as a key, which is associated with a datum or record and used to identify it to the data storage and retrieval application. The Hash function is applied to each value in the array to find its corresponding address in the address table.

#587 Explained Report Bookmark

If the number of record to be sorted large and the key is short, then …… sorting can be efficient.

null

- A
  Merge

- B
  Heap
- C
  Radix
- D
  Bubble

Correct Answer :C

Explanation

Radix sort is a sorting technique that sorts the elements by first grouping the individual digits of the same place value. Then, sort the elements according to their increasing/decreasing order.

#588 Explained Report Bookmark

If the number of record to be sorted large and the key is long, then …… sorting can be efficient.

null

- A
  Merge
- B
  Heap
- C
  Quick
- D
  Bubble

Correct Answer :C

Explanation

the Quick sort algorithm generally is the best for large data sets and long keys. This is because in the average case the time complexity is O(n logn). But, in sorting algorithms it's best that you always analise the type of data you will apply the algorithm and then decide wich one you gonna use.

#589 Explained Report Bookmark

The time complexity of heap sort is ….

null

- **A**
  O(n)
- **B**
  O(logn)
- **C**
  O(n2)
- **D**
  O(n logn)

Correct Answer :D

Explanation

overall time complexity of Heap Sort is O(N*LogN) where N is the number of elements in the list or array.

#590 Explained Report Bookmark

The worst case occur in linear search algorithm when …….

null

- **A**
  Item is somewhere in the middle of the array
- **B**
  Item is not in the array at all

- C
  Item is the last element in the array
- D
  Item is the last element in the array or item is not there at all

Correct Answer :D

Explanation

Algorithmic complexity is concerned about how fast or slow particular algorithm performs.

#591 Explained Report Bookmark

.If the number of records to be sorted is small, then …… sorting can be efficient.

null

- A
  Merge
- B
  Heap
- C
  Selection
- D
  Bubble

Correct Answer :C

Explanation

For small arrays (less than 20-30 elements), both insertion sort and selection sort are typically faster

#592 Explained Report Bookmark

The complexity of sorting algorithm measures the …… as a function of the number n of items to be sorter.

null

- A
  average time
- B
  running time
- C
  average-case complexity
- D
  case-complexity

Correct Answer :B

Explanation

it is a comparison based sorting algorithm, usually your goal is to count the number of comparisons made as that's the elementary operation typically, but if it has more involved pieces you have to account for those.

#593 Explained Report Bookmark

Which of the following is not a limitation of binary search algorithm?

null

- A
  must use a sorted array
- B
  requirement of sorted array is expensive when a lot of insertion and deletions are needed
- C
  there must be a mechanism to access middle element directly
- D
  binary search algorithm is not efficient when the data elements more than 1500

Correct Answer :D

Explanation

inary Search is applied on the sorted array or list of large size

#594 Explained Report Bookmark

The Average case occurs in linear search algorithm ……….

null

- A
  when item is somewhere in the middle of the array
- B
  when item is not the array at all
- C
  when item is the last element in the array

- D
  Item is the last element in the array or item is not there at all

Correct Answer :A

Explanation

number of comparisons will be around n/2. rest 3 choices are worst case where number of comparisons will be exactly n

#595 Explained Report Bookmark

Sorting algorithm can be characterized as ……

null

- A
  Simple algorithm which require the order of n2 comparisons to sort n items.
- B
  Sophisticated algorithms that require the O(nlog2n) comparisons to sort items.
- C
  Both of the above
- D
  None of the above

Correct Answer :C

Explanation

A and B both are true and the first method move data only over small distances in the process of sorting, whereas the second method method large distances, so that items settle into the proper order sooner, thus resulting in fewer comparisons. Performance of a sorting algorithm can also depend on the degree of order a heady present in the data.

#596 Explained Report Bookmark

…………order is the best possible for array sorting algorithm which sorts n item.

null

- A
  O(n logn)
- B
  O(n2)
- C
  O(n+logn)
- D
  O(logn)

Correct Answer :C

Explanation

O(log(n)) - logarithmic time - as input gets larger, so will the time, but by a decreasing amount, O(n*log(n)) - linear * logarithmic - increases larger than linear, but not as fast as the following, O(n^2), or generally O(n^k) where k is a constant - polynomial time, probably the worst of feasible algorithms, Judging by the above range of complexities, we can say that O(n+logn) is the best possible order.

#597 Explained Report Bookmark

………… is the method used by card sorter.

null

- A
  Radix sort
- B
  Insertion
- C
  Heap
- D
  Quick

Correct Answer :A

Explanation

Radix sort is one of the linear sorting algorithms for integers. It functions by sorting the input numbers on each digit, for each of the digits in the numbers. Radix sort can be applied to data that can be sorted lexicographically, be they integers, words, punch cards, playing cards, or the mail.

#598 Explained Report Bookmark

Which of the following sorting algorithm is of divide and conquer type?

null

- A
  Bubble sort
- B
  Insertion sort
- C
  Merge sort
- D
  Selection sort

Correct Answer :C

Explanation

Merge sort is a divide-and-conquer algorithm based on the idea of breaking down a list into several sub-lists until each sublist consists of a single element and merging those sublists in a manner that results into a sorted list. Idea: Divide the unsorted list into sublists, each containing element.

#599 Explained Report Bookmark

…….. sorting algorithm is frequently used when n is small where n is total number of elements.

null

- A
  Heap
- B
  Insertion
- C
  Bubble
- D
  Quick

Correct Answer :B

Explanation

non-recursive sorting algorithms such as insertion sort or selection sort are generally faster for very small arrays (the exact size varies by environment and implementation, but is typically around

#600 Explained Report Bookmark

Which of the following sorting algorithm is of priority queue sorting type?

null

- A
  Bubble sort
- B
   Insertion sort
- C
  Merge sort
- D
  Selection sort

Correct Answer :D

Explanation

Because at each step, it finds the minimum/maximum element.

#601 Explained Report Bookmark

Which of the following is not the required condition for binary search algorithm?

null

- A
  The list must be sorted
- B
  There should be the direct access to the middle element in any sub list
- C
  There must be mechanism to delete and/or insert elements in list.
- D
  Number values should only be present

Correct Answer :C

Explanation

Binary search mainly requires the middle first and last element in order to sort the list. Thus there is no necessary scope of having to delete or insert elements in a list.

#602 Explained Report Bookmark

Partition and exchange sort is ……..

null

- A
  quick sort
- B
   tree sort
- C
   heap sort
- D
  bubble sort

Correct Answer :A

Explanation

Quick sort is also called partition-exchange sort because of its working mechanism. This algorithm divides the list into three main parts

#603 Explained Report Bookmark

Binary trees with threads are called as …….

null

- A
  Threaded trees
- B
  Pointer trees
- C
  Special trees

- D
  Special pointer trees

Correct Answer :A

Explanation

Threaded Tree makes use of NULL pointers to improve its traversal process. In a threaded binary tree, NULL pointers are replaced by references of other nodes in the tree. These extra references are called as threads.

#604 Explained Report Bookmark

Graph G is ………….. if for any pair u, v of nodes in G there is a path from u to v or path from v to u.

null

- A
  Literally connected
- B
  Widely Connected
- C
  Unliterally connected
- D
  Literally connected

Correct Answer :C

Explanation

A graph is said to be unilaterally connected if it contains a directed path from u to v OR a directed path from v to u for every pair of vertices u, v. Hence, at least for any pair of vertices, one vertex should be reachable form the other. Such a path matrix would rather have upper triangle elements containing 1’s OR lower triangle elements containing 1’s

#605Explained Report Bookmark

In Binary trees nodes with no successor are called ……

null

- A
  End nodes
- B
  Terminal nodes
- C
  Final nodes
- D
  Last nodes

Correct Answer :B

Explanation

Leaf node, a node of a tree data structure that has no child nodes. Lymph node, a terminal lymph node in the lymphatic system.

#606Explained Report Bookmark

A connected graph T without any cycles is called ……..

null

- A
  free graph
- B
  no cycle graph
- C
  non cycle graph
- D
  circular graph

Correct Answer :A

Explanation

In graph theory, a tree is an undirected graph in which any two vertices are connected by exactly one path, or equivalently a connected acyclic undirected graph. Free graph is also called as a tree graph and a graph.

#607 Explained Report Bookmark

Trees are said ………. if they are similar and have same contents at corresponding nodes.

null

- A
  Duplicate
- B
  Carbon copy
- C
  Replica
- D
  Copies

Correct Answer :D

Explanation

tree data structure is a non-linear data structure because it does not store in a sequential manner. Thus corresponding node holding same value as current node is a plain copy of current node.

#608 Explained Report Bookmark

A connected graph T without any cycles is called a ……..

null

- A
 A tree graph
- B
 Free tree
- C
 A tree d
- D
 All of the above

Correct Answer :D

Explanation

In graph theory, a tree is an undirected graph in which any two vertices are connected by exactly one path, or equivalently a connected acyclic undirected graph. Free graph is also called as a tree graph and a graph.

#609 Explained Report Bookmark

Every node N in a binary tree T except the root has a unique parent called the ……… of N.

null

- A
 Antecedents
- B
 Predecessor
- C
 Forerunner
- D
 Precursor

Correct Answer :B

Explanation

predecessor is the maximum value in its left subtree and its successor the minimum value in its right subtree.

#610 Explained Report Bookmark

In a graph if E=(u,v) means ……

null

- A
  u is adjacent to v but v is not adjacent to u
- B
  e begins at u and ends at v
- C
  u is processor and v is successor
- D
  both b and c

Correct Answer :D

Explanation

graph is a common data structure that consists of a finite set of nodes (or vertices) and a set of edges connecting them. A pair (x,y) is referred to as an edge, which communicates that the x vertex connects to the y vertex.

#611 Explained Report Bookmark

Sequential representation of binary tree uses ……..

null

- A
  Array with pointers
- B
  Single linear array
- C
  Two dimensional arrays

- D
  Three dimensional arrays

Correct Answer :A

Explanation

all binary trees consist of a root pointer that points in the direction of the root node. When you see a root node pointing towards null or 0, you should know that you are dealing with an empty binary tree.

#612 Explained Report Bookmark

In a graph if e=[u,v], Then u and v are called ……..

null

- A
  End points of e
- B
  Adjacent nodes
- C
  . Neighbors
- D
  All of the above

Correct Answer :D

Explanation

graph is a common data structure that consists of a finite set of nodes (or vertices) and a set of edges connecting them. A pair (x,y) is referred to as an edge, which communicates that the x vertex connects to the y vertex.

#613 Explained Report Bookmark

TREE[1]=NULL indicates tree is ……..

null

- A
  Overflow
- B
  Underflow
- C
  Empty
- D
  Full

Correct Answer :C

Explanation

A tree is represented by a pointer to the topmost node in tree. If the tree is empty, then value of root is NULL.

#614 Explained Report Bookmark

A binary tree whose every node has either zero or two children is called …….

null

- A
  complete binary tree
- B
  binary search tree
- C
  extended binary tree
- D
  data structure

Correct Answer :C

Explanation

Extended binary tree consists of replacing every null subtree of the original tree with special nodes.

#615 Explained Report Bookmark

Linked representation of binary tree needs ……… parallel arrays.

null

- A
  4
- B
  2
- C
  3
- D
  5

Correct Answer :C

Explanation

We use a double linked list to represent a binary tree. In a double linked list, every node consists of three fields. First field for storing left child address, second for storing actual data and third for storing right child address. Thus we require 3 parallel arrays to represent binary tree in linked form.

#616 Explained Report Bookmark

The depth of complete binary tree is given by ……

null

- A
  Dn = n log2n

- B
  Dn= n log2n+1
- C
  Dn = log2n
- D
  Dn = log2n+1

Correct Answer :D

Explanation

The depth of complete binary tree of n nodes will be Dn=log 2 (n+1). Here Dn is the height or depth of the tree and n is the number of nodes. A complete binary tree is a binary tree where all the levels have maximum number of nodes except possibly the last level. There is a difference between complete binary tree and full binary tree. A binary tree is called as full binary in which every node other than the leaves has two children. It is also called as 2-tree.

#617 Explained Report Bookmark

Which indicates pre-order traversal?

null

- A
  Left sub-tree, Right sub-tree and root
- B
  Right sub-tree, Left sub-tree and root
- C
  Root, Left sub-tree, Right sub-tree
- D
  Right sub-tree, root, Left sub-tree

Correct Answer :C

Explanation

In PreOrder traversal, the root is visited first, followed by left subtree and the right subtree, hence it is also known as NLR (nod-left-right) algorithm as well.

#618 Explained Report Bookmark

A terminal node in a binary tree is called …………

null

- A
  Root
- B
  Leaf
- C
  Child
- D
  Branch

Correct Answer :B

Explanation

A node with degree zero is call a terminal node or a leaf.

#619 Explained Report Bookmark

The post order traversal of binary tree is DEBFCA. Find out the pre order traversal.

null

- A
  ABFCDE
- B
  ADBFEC
- C
  ABDECF
- D
  ABDCEF

Correct Answer :C

Explanation

Step 1: Construct binary tree from the given post order traversal : debfca.

1.From the given post order traversal will identify that “a” is the root.

2. “a” being the root has a left and right side. Since left side is considered first, it is expected that it will have two sides. Hence, “a” will have “deb” on the left node and “fc” on the right.
3. Further on the left side, we understand that “b” is the root, which will have left side as “d” and right one as “e”.
4. On the right side of “a”, c will be the root which will only have a left side with “f”
5. Hence, below binary tree is constructed:

Step 2: Perform pre-order traversal of the constructed binary tree.

After traversing the above tree in the pre-order, we get: abdecf.

#620 Explained Report Bookmark

While converting binary tree into extended binary tree, all the original nodes in binary tree are …….

null

- A
  Internal nodes on extended tree
- B
  External nodes on extended tree
- C
  Vanished on extended tree
- D
  Intermediate nodes on extended tree

Correct Answer :A

Explanation

The nodes from the original tree are internal nodes and the special nodes are external nodes. All external nodes are leaf nodes and the internal nodes are non-leaf nodes. Every internal node has exactly two children and every external node is a leaf. It displays the result which is a complete binary tree

#621 Explained Report Bookmark

In a binary tree, certain null entries are replaced by special pointers which point to nodes higher in the tree for efficiency. These special pointers are called ....

null

- A
  Leaf
- B
  Branch
- C
  Branch
- D
  Thread

Correct Answer :D

Explanation

The idea of threaded binary trees is to make inorder traversal faster and do it without stack and without recursion. A binary tree is made threaded by making all right child pointers that would normally be NULL point to the inorder successor of the node (if it exists).

#622 Explained Report Bookmark

A binary tree whose every node has either zero or two children is called ………

null

- A
  Complete binary tree
- B
  . Binary Search tree
- C
  Extended binary tree
- D
  E2 tree

Correct Answer :C

Explanation

A binary tree whose every node has either zero or two children is called extended binary tree. Extended binary tree consists of replacing every null subtree of the original tree with special nodes. Empty circle represents internal node and filled circle represents external node. The nodes from the original tree are internal nodes and the special nodes are external nodes. Every internal node in the extended binary tree has exactly two children and every external node is a leaf. It displays the result which is a complete binary tree.

#623 Explained Report Bookmark

If every node u in G is adjacent to every other node v in G,A graph is said to be ……..

null

- A
  isolated
- B
  complete
- C
  finite

- D
  strongly connected.

Correct Answer :B

Explanation

A complete graph is a graph in which each pair of graph vertices is connected by an edge

#624 Explained Report Bookmark

Three standards ways of traversing a binary tree T with root R …….

null

- A
  Prefix, infix, postfix
- B
  Pre-process, in-process, post-process
- C
  Pre-traversal, in-traversal, post-traversal
- D
  Pre-order, in-order, post-order

Correct Answer :D

Explanation

Inorder Traversal: For binary search trees (BST), Inorder Traversal specifies the nodes in non-descending order. In order to obtain nodes from BST in non-increasing order, a variation of inorder traversal may be used where inorder traversal is reversed.

Preorder Traversal: Preorder traversal will create a copy of the tree. Preorder Traversal is also used to get the prefix expression of an expression.

Postorder Traversal: Postorder traversal is used to get the postfix expression of an expression given

#625 Explained Report Bookmark

A graph is said to be ……. if every node u in G is adjacent to every other node v in G.

null

- A
  Absolute
- B
  Entire
- C
  Entire
- D
  Complete

Correct Answer :D

Explanation

A complete graph is a graph with N vertices and an edge between every two vertices. There are no loops and every two vertices share exactly one edge.

#626 Explained Report Bookmark

A graph is said to be ……. if its edges are assigned data.

null

- A
  Tagged

- B
  Marked
- C
  Lebeled
- D
  Sticked

Correct Answer :C

Explanation

A labeled graph is a graph whose vertices are each assigned an element from a set of symbols

#627 Explained Report Bookmark

If node N is a terminal node in a binary tree then its ………

null

- A
  . Right tree is empty
- B
  Left tree is empty
- C
  Both left & right sub trees are empty
- D
  Root node is empty

Correct Answer :C

Explanation

terminal node is also called as leaf node i.e. a node of a tree data structure that has no child nodes.

#628 Explained Report Bookmark

Quick sort uses ____ for implementation

- A
  traversal
- B
  heaps
- C
  queues
- D
  recursion

Correct Answer :D

Explanation

Quick sort is a divide and conquer algorithm. Quick sort first partitions a large array into two smaller sub-arrays. And then recursively sorts the sub-arrays.

#629 Explained Report Bookmark

Arrays are best data structures

- A
  for relatively permanent collections of data
- B
  for the size of the structure and the data in the structure are constantly changing
- C
  for both of above situation
- D
  for none of above situation

Correct Answer :A

Explanation

The operator tree has a tree like format where the evaluation starts from root of the tree.

#630 Explained Report Bookmark

The organization and management of data structures are take place in:

null

- A
  Primary Memory
- B
  Secondary Memory
- C
  External Memory
- D
  Primary & Secondary Memory

Correct Answer :A

Explanation

a data structure is a data organization, management, and storage format that enables efficient access and modification through primary memory.

#631 Explained Report Bookmark

The worst case height of an AVL tree with n nodes is ……………

null

- A
  2 lg n
- B
  1.39 lg n

- C
  1.44 lg n
- D
  1.64 lg

Correct Answer :C

Explanation

The worst case possible height of AVL tree with n nodes is 1.44*logn. This can be verified using AVL tree having 7 nodes and maximum height.

#632Explained Report Bookmark

While deleting nodes from a binary heap, …………………. node is replaced by the last leaf in the tree.

null

- A
  left leaf
- B
  right leaf
- C
  root
- D
  cycle

Correct Answer :C

Explanation

Deleting nodes from a binary heap, root node is replaced by the last leaf in the tree.

#633Explained Report Bookmark

An insertion into a ……………………….. is performed by inserting the new node in the location referenced by next in the array and then "sifting it up" by comparing the key of the newly inserted node with the key of the parent.

null

- A
  red-black
- B
  AVL
- C
  binary search
- D
   binary heap

Correct Answer :D

Explanation

It's a complete tree (All levels are completely filled except possibly the last level and the last level has all keys as left as possible). This property of Binary Heap makes them suitable to be stored in an array.

2) A Binary Heap is either Min Heap or Max Heap. In a Min Binary Heap, the key at root must be minimum among all keys present in Binary Heap. The same property must be recursively true for all nodes in Binary Tree. Max Binary Heap is similar to MinHeap.

#634 Explained Report Bookmark

In binary search tree, a …………………… rooted to node n is the tree formed by imaging node n was a root.

null

- A
  cycle

- B
  node
- C
  root
- D
  subtree

Correct Answer :D

Explanation

Binary Search Tree (BST) Every element in n's left subtree is less than or equal to the element in node n.

#635 Explained Report Bookmark

Which of the following statements for a simple graph is correct?

null

- 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

Correct Answer :A

Explanation

In a walk if the vertices are distinct it is called a path, whereas if the edges are distinct it is called a trail.

#636 Explained Report Bookmark

Which of the following is the most widely used external memory data structure?

null

- **A**
  AVL tree
- **B**
  B-tree
- **C**
  Red-black tree
- **D**
  Both AVL tree and Red-black tree

Correct Answer :B

Explanation

In external memory, the data is transferred in form of blocks. These blocks have data valued and pointers. And B-tree can hold both the data values and pointers. So B-tree is used as an external memory data structure.

#637 Explained Report Bookmark

What is an AVL tree?

null

- **A**
  a tree which is balanced and is a height balanced tree
- **B**
  a tree which is unbalanced and is a height balanced tree
- **C**
  a tree with three children
- **D**
  a tree with atmost 3 children

Correct Answer :A

Explanation

It is a self balancing tree with height difference atmost 1.

#638 Explained Report Bookmark

The binary tree sort implemented using a self – balancing binary search tree takes ______ time is worst case.

null

- A
  O(n log n)
- B
  O(n)
- C
  O(n2)
- D
  O(log n)

Correct Answer :A

Explanation

 The worst case running time of the binary tree sort is O(n2). But, the worst case running time can be improved to the O(n log n) using a self – balancing binary search tree.

#639 Explained Report Bookmark

A self – balancing binary search tree can be used to implement ________

null

- A
  Priority queue
- B
  Hash table
- C
  Heap sort
- D
  Priority queue and Heap sort

Correct Answer :A

Explanation

Self-balancing binary search trees can be used to construct and maintain ordered lists, to achieve the optimal worst case performance. So, self – balancing binary search tree can be used to implement a priority queue, which is ordered list.

#640 Explained Report Bookmark

What is the Height of the root node of ternary tree?

null

- A
  1
- B
  2
- C
  3
- D
  0

Correct Answer :D

Explanation

Height of ternary tree is defined as the length of path from root to deepest node in tree. Therefore, height off root node in ternary tree is 0.