**2013**

Passing out in 2013, then you are Awesome



BASIC DATA STRUCTURE FOR YOU

#### LIST OF IMPORTANT DATA STRUCTURE QUESTIONS (BASICS)

# What is a data structure? What are the types of data structures? Briefly explain them

The scheme of organizing related information is known as ‘data structure’. The types of data structure are:

**Lists:** A group of similar items with connectivity to the previous or/and next data items.  
**Arrays:** A set of homogeneous values  
Records: A set of fields, where each field consists of data belongs to one data type.  
**Trees:** A data structure where the data is organized in a hierarchical structure. This type of data structure follows the sorted order of insertion, deletion and modification of data items.  
**Tables:** Data is persisted in the form of rows and columns. These are similar to records, where the result or manipulation of data is reflected for the whole table.

**1. Define a linear and nonlinear data structure.**

**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

**Non-Linear data structure:** Every data item is attached to several other data items in a way that is specific for reflecting relationships. The data items are not arranged in a sequential structure. Ex: Trees, Graphs

**2. Define in brief an array. What are the types of array operations?**

An array is a set of homogeneous elements. Every element is referred by an index.

Arrays are used for storing the data until the application expires in the main memory of the computer system. So that, the elements can be accessed at any time. The operations are:

- Adding elements  
- Sorting elements  
- Searching elements  
- Re-arranging the elements   
- Performing matrix operations  
- Pre-fix and post-fix operations

**3. What is a matrix? Explain its uses with an example**

A matrix is a representation of certain rows and columns, to persist homogeneous data. It can also be called as double-dimensioned array.

Uses:  
  
- To represent class hierarchy using Boolean square matrix  
- For data encryption and decryption  
- To represent traffic flow and plumbing in a network  
- To implement graph theory of node representation

**4 . Define an algorithm. What are the properties of an algorithm? What are the types of algorithms?**

**Algorithm:** A step by step process to get the solution for a well-defined problem.  
**Properties of an algorithm:**

- Should be written in simple English  
- Should be unambiguous, precise and lucid  
- Should provide the correct solutions   
- Should have an end point  
- The output statements should follow input, process instructions  
- The initial statements should be of input statements  
- Should have finite number of steps  
- Every statement should be definitive

**Types of algorithms:**

- Simple recursive algorithms. Ex: Searching an element in a list  
– Backtracking algorithms Ex: Depth-first recursive search in a tree  
– Divide and conquer algorithms. Ex: Quick sort and merge sort  
– Dynamic programming algorithms. Ex: Generation of Fibonacci series  
– Greedy algorithms Ex: Counting currency   
– Branch and bound algorithms. Ex: Travelling salesman (visiting each city once and minimize the total distance travelled)  
– Brute force algorithms. Ex: Finding the best path for a travelling salesman  
– Randomized algorithms. Ex. Using a random number to choose a pivot in quick sort).

**5. What is an iterative algorithm?**

The process of attempting for solving a problem which finds successive approximations for solution, starting from an initial guess. The result of repeated calculations is a sequence of approximate values for the quantities of interest.

**6 . What is a recursive algorithm?**

Recursive algorithm is a method of simplification that divides the problem into sub-problems of the same nature. The result of one recursion is the input for the next recursion. The repletion is in the self-similar fashion. The algorithm calls itself with smaller input values and obtains the results by simply performing the operations on these smaller values. Generation of factorial, Fibonacci number series are the examples of recursive algorithms.

**7 . Explain quick sort and merge sort algorithms**.

Quick sort employs the ‘divide and conquer’ concept by dividing the list of elements into two sub elements.

The process is as follows:

1. Select an element, pivot, from the list.  
2. Rearrange the elements in the list, so that all elements those are less than the pivot are arranged before the pivot and all elements those are greater than the pivot are arranged after the pivot. Now the pivot is in its position.  
3. Sort the both sub lists – sub list of the elements which are less than the pivot and the list of elements which are more than the pivot recursively.

Merge Sort: A comparison based sorting algorithm. The input order is preserved in the sorted output.

Merge Sort algorithm is as follows:

1. The length of the list is 0 or 1, and then it is considered as sorted.  
2. Otherwise, divide the unsorted list into 2 lists each about half the size.  
3. Sort each sub list recursively. Implement the step 2 until the two sub lists are sorted.  
4. As a final step, combine (merge) both the lists back into one sorted list.

**8 . What is Bubble Sort and Quick sort?**

**Bubble Sort:** The simplest sorting algorithm. It involves the sorting the list in a repetitive fashion. It compares two adjacent elements in the list, and swaps them if they are not in the designated order. It continues until there are no swaps needed. This is the signal for the list that is sorted. It is also called as comparison sort as it uses comparisons.

**Quick Sort:** The best sorting algorithm which implements the ‘divide and conquer’ concept. It first divides the list into two parts by picking an element a ’pivot’. It then arranges the elements those are smaller than pivot into one sub list and the elements those are greater than pivot into one sub list by keeping the pivot in its original place.

**9 . What are the difference between a stack and a Queue?**

**Stack** – Represents the collection of elements in Last In First Out order.   
Operations includes testing null stack, finding the top element in the stack, removal of top most element and adding elements on the top of the stack.

**Queue** - Represents the collection of elements in First In First Out order.

Operations include testing null queue, finding the next element, removal of elements and inserting the elements from the queue.

Insertion of elements is at the end of the queue

Deletion of elements is from the beginning of the queue.

**10 . Can a stack be described as a pointer? Explain.**

A stack is represented as a pointer. The reason is that, it has a head pointer which points to the top of the stack. The stack operations are performed using the head pointer. Hence, the stack can be described as a pointer.

**11.. Explain the terms Base case, Recursive case, Binding Time, Run-Time Stack and Tail Recursion.**

**Base case:** A case in recursion, in which the answer is known when the termination for a recursive condition is to unwind back.

**Recursive Case:** A case which returns to the answer which is closer.

**Run-time Stack:** A run time stack used for saving the frame stack of a function when every recursion or every call occurs.

**Tail Recursion:** It is a situation where a single recursive call is consisted by a function, and it is the final statement to be executed. It can be replaced by iteration.

**12 . Is it possible to insert different type of elements in a stack? How?**

Different elements can be inserted into a stack. This is possible by implementing union / structure data type. It is efficient to use union rather than structure, as only one item’s memory is used at a time.

**13 . Explain in brief a linked list.**

A linked list is a dynamic data structure. It consists of a sequence of data elements and a reference to the next record in the sequence. Stacks, queues, hash tables, linear equations, prefix and post fix operations. The order of linked items is different that of arrays. The insertion or deletion operations are constant in number.

**14 . Explain the types of linked lists.**

The types of linked lists are:

**Singly linked list:** It has only head part and corresponding references to the next nodes.

**Doubly linked list:** A linked list which both head and tail parts, thus allowing the traversal in bi-directional fashion. Except the first node, the head node refers to the previous node.

**Circular linked list:** A linked list whose last node has reference to the first node.

**15. . How would you sort a linked list?**

**Step 1:** Compare the current node in the unsorted list with every element in the rest of the list. If the current element is more than any other element go to step 2 otherwise go to step 3.

**Step 2:** Position the element with higher value after the position of the current element. Compare the next element. Go to step1 if an element exists, else stop the process.

**Step 3:** If the list is already in sorted order, insert the current node at the end of the list. Compare the next element, if any and go to step 1 or quit.

**16 . What is sequential search? What is the average number of comparisons in a sequential search?**

**Sequential search:** Searching an element in an array, the search starts from the first element till the last element.

The average number of comparisons in a sequential search is (N+1)/2 where N is the size of the array. If the element is in the 1st position, the number of comparisons will be 1 and if the element is in the last position, the number of comparisons will be N.

**17 . What are binary search and Fibonacci search?**

**Binary Search:** Binary search is the process of locating an element in a sorted list. The search starts by dividing the list into two parts. The algorithm compares the median value. If the search element is less than the median value, the top list only will be searched, after finding the middle element of that list. The process continues until the element is found or the search in the top list is completed. The same process is continued for the bottom list, until the element is found or the search in the bottom list is completed. If an element is found that must be the median value.

**Fibonacci Search:** Fibonacci search is a process of searching a sorted array by utilizing divide and conquer algorithm. Fibonacci search examines locations whose addresses have lower dispersion. When the search element has non-uniform access memory storage, the Fibonacci search algorithm reduces the average time needed for accessing a storage location.

## 18. Which data structure is used to perform recursion?

- The data structure used for recursion is Stack.  
- Its LIFO property helps it remembers its 'caller'. This helps it know the data which is to be returned when the function has to return.  
- System stack is used for storing the return addresses of the function calls.

## 19. Does the Minimal Spanning tree of a graph give the shortest distance between any 2 specified nodes?

- No, it doesn’t.  
- It assures that the total weight of the tree is kept to minimum.  
- It doesn't imply that the distance between any two nodes involved in the minimum-spanning tree is minimum.

## 20. If you are using C language to implement the heterogeneous linked list, what pointer type will you use?

- A heterogeneous linked list contains different data types in its nodes. We can not use ordinary pointer to connect them.   
- The pointer that we use in such a case is void pointer as it is a generic pointer type and capable of storing pointer to any type.

## 21. Differentiate between PUSH and POP?

- Pushing and popping refers to the way data is stored into and retrieved from a stack.  
- PUSH – Data being pushed/ added to the stack.  
- POP - Data being retrieved from the stack, particularly the topmost data.

## 22. When is a binary search algorithm best applied?

- It is best applied to search a list when the elements are already in order or sorted.  
- The list here is searched starting in the middle. If that middle value is not the correct one, the lower or the upper half is searched in the similar way.

## 23. How do you reference all the elements in a one-dimension array?

- This is done using an indexed loop.   
- The counter runs from 0 to the array size minus one.   
- Using the loop counter as the array subscript helps in referencing all the elements in one-dimensional array.

## 24. What is Huffman’s algorithm?

- It is used in creating extended binary trees that have minimum weighted path lengths from the given weights.  
- It makes use of a table that contains frequency of occurrence for each data element.

## 25. What is Fibonacci search?

- It is a search algorithm that applies to a sorted array.  
- It uses divide-and-conquer approach that reduces the time needed to reach the target element.

## 26. Which data structure is applied when dealing with a recursive function?

- A recursive function is a function that calls itself based on a terminating condition.   
- It uses stack.   
- Using LIFO, a call to a recursive function saves the return address. This tells the return address to the calling function after the call terminates.

## 27. How does dynamic memory allocation help in managing data?

- Dynamic memory allocation helps to store simple structured data types.   
- It can combine separately allocated structured blocks to form composite structures that expand and contract as required.

## 28. What is a bubble sort and how do you perform it?

- Bubble sort is a sorting technique which can be applied to data structures like arrays.  
- Here, the adjacent values are compared and their positions are exchanged if they are out of order.   
- The smaller value bubbles up to the top of the list, while the larger value sinks to the bottom.

## 29. How does variable declaration affect memory allocation?

- The amount of memory to be allocated depends on the data type of the variable.  
- An integer type variable is needs 32 bits of memory storage to be reserved.

## 30. You want to insert a new item in a binary search tree. How would you do it?

- Let us assume that the you want to insert is unique.  
- First of all, check if the tree is empty.   
- If it is empty, you can insert the new item in the root node.   
- If it is not empty, refer to the new item’s key.   
- If the data to be entered is smaller than the root’s key, insert it into the root’s left subtree.   
- Otherwise, insert it into the root’s right subtree.

## 31. Why is the isEmpty() member method called?

- The isEmpty() member method is called during the dequeue process. It helps in ascertaining if there exists any item in the queue which needs to be removed.   
- This method is called by the dequeue() method before returning the front element.

## 32. What is a queue?

- A Queue refers to a sequential organization of data.   
- It is a FIFO type data structure in which an element is always inserted at the last position and any element is always removed from the first position.

## 33. What is a DE queue?

- A DE queue is a double-ended queue.  
- The elements here can be inserted or removed from either end.

## 34. What is a postfix expression?

- It is an expression in which each operator follows its operands.  
- Here, there is no need to group sub-expressions in parentheses or to consider operator precedence..

\*\*PS : IF you find this useful and our response was appreciable then Please promote our page by sharing it’s link on your wall and asking your friends to LIKE this.