



This Download is from www.downloadmela.com . The main motto of this website is to provide free download links of ebooks,video tutorials,magazines,previous papers,interview related content. To download more visit the website.

If you like our services please help us in 2 ways.

1.Donate money.

Please go through the link to donate

<http://www.downloadmela.com/donate.html>

2.Tell about this website to your friends,relatives.

Thanks for downloading. Enjoy the reading.

Visit <http://www.downloadmela.com/> for more papers

What is data structure?

A data structure is a way of organizing data that considers not only the items stored, but also their relationship to each other. Advance knowledge about the relationship between data items allows designing of efficient algorithms for the manipulation of data.

List out the areas in which data structures are applied extensively?

Compiler Design, Operating System, Database Management System, Statistical analysis package, Numerical Analysis, Graphics, Artificial Intelligence, Simulation

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

The heterogeneous linked list contains different data types in its nodes and we need a link, pointer to connect them. It is not possible to use ordinary pointers for this. So we go for void pointer. Void pointer is capable of storing pointer to any type as it is a generic pointer type.

What is the data structures used to perform recursion?

Stack. Because of its LIFO (Last In First Out) property it remembers its caller, so knows whom to return when the function has to return. Recursion makes use of system stack for storing the return addresses of the function calls. Every recursive function has its equivalent iterative (non-recursive) function. Even when such equivalent iterative procedures are written, explicit stack is to be used.

What are the methods available in storing sequential files ?

Straight merging, Natural merging, Polyphase sort, Distribution of Initial runs.

List out few of the Application of tree data-structure?

The manipulation of Arithmetic expression, Symbol Table construction, Syntax analysis.

In RDBMS, what is the efficient data structure used in the internal storage representation?

B+ tree. Because in B+ tree, all the data is stored only in leaf nodes, that makes searching easier. This corresponds to the records that shall be stored in leaf nodes.

What is a spanning Tree?

A spanning tree is a tree associated with a network. All the nodes of the graph appear on the tree once. A minimum spanning tree is a spanning tree organized so that the total edge weight between nodes is minimized.

Does the minimum spanning tree of a graph give the shortest distance between any 2 specified nodes?

Minimal spanning tree assures that the total weight of the tree is kept at its minimum. But it doesn't mean that the distance between any two nodes involved in the minimum-spanning tree is minimum.

Whether Linked List is linear or Non-linear data structure?

According to Access strategies Linked list is a linear one. According to Storage Linked List is a Non-linear one.

What is the quickest sorting method to use?

The answer depends on what you mean by quickest. For most sorting problems, it just doesn't matter how quick the sort is because it is done infrequently or other operations take significantly more time anyway. Even in cases in which sorting speed is of the essence, there is no one answer. It depends on not only the size and nature of the data, but also the likely order. No algorithm is best in all cases.

There are three sorting methods in this author's toolbox that are all very fast and that are useful in different situations. Those methods are quick sort, merge sort, and radix sort.

The Quick Sort

The quick sort algorithm is of the divide and conquer type. That means it works by reducing a sorting problem into several easier sorting problems and solving each of them. A dividing value is chosen from the input data, and the data is partitioned into three sets: elements that belong before the dividing value, the value itself, and elements that come after the dividing value. The partitioning is performed by exchanging elements that are in the first set but belong in the third with elements that are in the third set but belong in the first. Elements that are equal to the dividing element can be put in any of the three sets the algorithm will still work properly.

The Merge Sort

The merge sort is a divide and conquer sort as well. It works by considering the data to be sorted as a sequence of already-sorted lists (in the worst case, each list is one element long). Adjacent sorted lists are merged into larger sorted lists until there is a single sorted list containing all the elements. The merge sort is good at sorting lists and other data structures that are not in arrays, and it can be used to sort things that don't fit into memory. It also can be implemented as a stable sort.

The Radix Sort

The radix sort takes a list of integers and puts each element on a smaller list, depending on the value of its least significant byte. Then the small lists are concatenated, and the process is repeated for each more significant byte until the list is sorted. The radix sort is simpler to implement on fixed-length data such as ints.

How can I search for data in a linked list?

Unfortunately, the only way to search a linked list is with a linear search, because the only way a linked list's members can be accessed is sequentially. Sometimes it is quicker to take the data from a linked list and store it in a different data structure so that searches can be more efficient.

What is the heap?

The heap is where malloc(), calloc(), and realloc() get memory.

Getting memory from the heap is much slower than getting it from the stack. On the other hand, the heap is much more flexible than the stack. Memory can be allocated at any time and deallocated in any order. Such memory isn't deallocated automatically; you have to call free().

Recursive data structures are almost always implemented with memory from the heap. Strings often come from there too, especially strings that could be very long at runtime. If you can keep data in a local variable (and allocate it from the stack), your code will run faster than if you put the data on the heap. Sometimes you can use a better algorithm if you use the heap faster, or more robust, or more flexible. It's a tradeoff.

If memory is allocated from the heap, it's available until the program ends. That's great if you remember to deallocate it when you're done. If you forget, it's a problem. A \square memory leak is some allocated memory that's no longer needed but isn't deallocated. If you have a memory leak inside a loop, you can use up all the memory on the heap and not be able to get any more. (When that happens, the allocation functions return a null pointer.) In some environments, if a program doesn't deallocate everything it

allocated, memory stays unavailable even after the program ends.

What is the easiest sorting method to use?

The answer is the standard library function `qsort()`. It's the easiest sort by far for several reasons:

It is already written.

It is already debugged.

It has been optimized as much as possible (usually).

`Void qsort(void *buf, size_t num, size_t size, int (*comp)(const void *ele1, const void *ele2));`

What is the bucket size, when the overlapping and collision occur at same time?

One. If there is only one entry possible in the bucket, when the collision occurs, there is no way to accommodate the colliding value. This results in the overlapping of values.

In an AVL tree, at what condition the balancing is to be done?

If the pivotal value (or the Height factor) is greater than 1 or less than 1.

Minimum number of queues needed to implement the priority queue?

Two. One queue is used for actual storing of data and another for storing priorities.

How many different trees are possible with 10 nodes ?

1014 - For example, consider a tree with 3 nodes($n=3$), it will have the maximum combination of 5 different (ie, $2^3 - 3 = 5$) trees.

What is a node class?

A node class is a class that, relies on the base class for services and implementation, provides a wider interface to users than its base class, relies primarily on virtual functions in its public interface depends on all its direct and indirect base class can be understood only in the context of the base class can be used as base for further derivation

can be used to create objects. A node class is a class that has added new services or functionality beyond the services inherited from its base class.

When can you tell that a memory leak will occur?

A memory leak occurs when a program loses the ability to free a block of dynamically allocated memory.

What is placement new?

When you want to call a constructor directly, you use the placement new. Sometimes you have some raw memory that's already been allocated, and you need to construct an object in the memory you have. Operator new's special version placement new allows you to do it.

```
class Widget
{
public :
Widget(int widgetsize);
...
Widget* Construct_widget_int_buffer(void *buffer,int widgetsize)
{
return new(buffer) Widget(widgetsize);
}
};
```

This function returns a pointer to a Widget object that's constructed within the buffer passed to the function. Such a function might be useful for applications using shared memory or memory-mapped I/

O, because objects in such applications must be placed at specific addresses or in memory allocated by special routines.

List out the areas in which data structures are applied extensively ?

Compiler Design, Operating System, Database Management System, Statistical analysis package, Numerical Analysis, Graphics, Artificial Intelligence, Simulation

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

The heterogeneous linked list contains different data types in its nodes and we need a link, pointer to connect them. It is not possible to use ordinary pointers for this. So we go for void pointer. Void pointer is capable of storing pointer to any type as it is a generic pointer type.

What is the data structures used to perform recursion?

Stack. Because of its LIFO (Last In First Out) property it remembers its caller so knows whom to return when the function has to return. Recursion makes use of system stack for storing the return addresses of the function calls. Every recursive function has its equivalent iterative (non-recursive) function. Even when such equivalent iterative procedures are written, explicit stack is to be used.

Whether Linked List is linear or Non-linear data structure?

According to Access strategies Linked list is a linear one.

According to Storage Linked List is a Non-linear one

Tell how to check whether a linked list is circular ?

Create two pointers, each set to the start of the list. Update each as follows:

```
while (pointer1)
{
pointer1 = pointer1->next;
pointer2 = pointer2->next; if (pointer2) pointer2=pointer2->next;
if (pointer1 == pointer2)
????? {
print ("\ncircular\n");
}
}
```

What is the difference between ARRAY and STACK?

STACK follows LIFO. Thus the item that is first entered would be the last removed.

In array the items can be entered or removed in any order. Basically each member access is done using index. No strict order is to be followed here to remove a particular element.

What is the difference between NULL AND VOID pointer?

NULL can be value for pointer type variables.
VOID is a type identifier which has not size.
NULL and void are not same. Example: void* ptr = NULL;

What is precision?

Precision refers to the accuracy of the decimal portion of a value. Precision is the number of digits allowed after the decimal point.

What is impact of signed numbers on the memory?

Sign of the number is the first bit of the storage allocated for that number. So you get one bit less for storing the number. For example if you are storing an 8-bit number, without sign, the range is 0-255. If you decide to store sign you get 7 bits for the number plus one bit for the sign. So the range is -128 to +127.

How memory is reserved using a declaration statement ?

Memory is reserved using data type in the variable declaration. A programming language implementation has predefined sizes for its data types.

For example, in C# the declaration `int i;` will reserve 32 bits for variable `i`.

A pointer declaration reserves memory for the address or the pointer variable, but not for the data that it will point to. The memory for the data pointed to by a pointer has to be allocated at runtime.

The memory reserved by the compiler for simple variables and for storing pointer address is allocated on the stack, while the memory allocated for pointer referenced data at runtime is allocated on the heap.

How many parts are there in a declaration statement?

There are two main parts, variable identifier and data type and the third type is optional which is type qualifier like signed/unsigned.

Is Pointer a variable?

Yes, a pointer is a variable and can be used as an element of a structure and as an attribute of a class in some programming languages such as C++, but not Java. However, the contents of a pointer is a memory address of another location of memory, which is usually the memory address of another variable, element of a structure, or attribute of a class.

What is Data Structure?

A data structure is a group of data elements grouped together under one name. These data elements, known as members, can have different types and different lengths. Some are used to store the data of same type and some are used to store different types of data.

What is significance of " * " ?

The symbol "*" tells the computer that you are declaring a pointer.

Actually it depends on context.

In a statement like `int *ptr;` the "*" tells that you are declaring a pointer.

In a statement like `int i = *ptr;` it tells that you want to assign value pointed to by `ptr` to variable `i`.

The symbol "*" is also called as Indirection Operator/ Dereferencing Operator.

Why do we Use a Multidimensional Array?

A multidimensional array can be useful to organize subgroups of data within an array. In addition to organizing data stored in elements of an array, a multidimensional array can store memory addresses of data in a pointer array and an array of pointers.

Multidimensional arrays are used to store information in a matrix form.

e.g. a railway timetable, schedule cannot be stored as a single dimensional array.
One can use a 3-D array for storing height, width and length of each room on each floor of a building.

How do you assign an address to an element of a pointer array ?

We can assign a memory address to an element of a pointer array by using the address operator, which is the ampersand (&), in an assignment statement such as `pemployee[0] = &projects[2];`

Run Time Memory Allocation is known as ?

Allocating memory at runtime is called a dynamically allocating memory. In this, you dynamically allocate memory by using the new operator when declaring the array, for example : `int grades[] = new int[10];`

What method is used to place a value onto the top of a stack?

`push()` method, Push is the direction that data is being added to the stack. `push()` member method places a value onto the top of a stack.

What method removes the value from the top of a stack?

The `pop()` member method removes the value from the top of a stack, which is then returned by the `pop()` member method to the statement that calls the `pop()` member method.

What does `isEmpty()` member method determines?

`isEmpty()` checks if the stack has at least one element. This method is called by `Pop()` before retrieving and returning the top element.

What is a queue ?

A Queue is a sequential organization of data. A queue is a first in first out type of data structure. An element is inserted at the last position and an element is always taken out from the first position.

What is the relationship between a queue and its underlying array?

Data stored in a queue is actually stored in an array. Two indexes, front and end will be used to identify the start and end of the queue.

When an element is removed front will be incremented by 1. In case it reaches past the last index available it will be reset to 0. Then it will be checked with end. If it is greater than end queue is empty.

When an element is added end will be incremented by 1. In case it reaches past the last index available it will be reset to 0. After incrementing it will be checked with front. If they are equal queue is full.

Which process places data at the back of the queue?

`Enqueue` is the process that places data at the back of the queue.

Why is the `isEmpty()` member method called?

The `isEmpty()` member method is called within the `dequeue` process to determine if there is an item in the queue to be removed i.e. `isEmpty()` is called to decide whether the queue has at least one element. This method is called by the `dequeue()` method before returning the front element.

How is the front of the queue calculated ?

The front of the queue is calculated by `front = (front+1) % size`.

What does each entry in the Link List called?

Each entry in a linked list is called a node. Think of a node as an entry that has three sub entries. One sub entry contains the data, which may be one attribute or many attributes. Another points to the

previous node, and the last points to the next node. When you enter a new item on a linked list, you allocate the new node and then set the pointers to previous and next nodes.

What is Linked List ?

Linked List is one of the fundamental data structures. It consists of a sequence of nodes, each containing arbitrary data fields and one or two ("links") pointing to the next and/or previous nodes. A linked list is a self-referential datatype because it contains a pointer or link to another data of the same type. Linked lists permit insertion and removal of nodes at any point in the list in constant time, but do not allow random access.

What member function places a new node at the end of the linked list?

The appendNode() member function places a new node at the end of the linked list. The appendNode() requires an integer representing the current data of the node.

How is any Data Structure application is classified among files?

A linked list application can be organized into a header file, source file and main application file. The first file is the header file that contains the definition of the NODE structure and the LinkedList class definition. The second file is a source code file containing the implementation of member functions of the LinkedList class. The last file is the application file that contains code that creates and uses the LinkedList class.

Which file contains the definition of member functions?

Definitions of member functions for the Linked List class are contained in the LinkedList.cpp file.

What are the major data structures used in the following areas : RDBMS, Network data model & Hierarchical data model.

1. RDBMS Array (i.e. Array of structures)
2. Network data model Graph
3. Hierarchical data model Trees.

Difference between calloc and malloc ?

malloc: allocate n bytes

calloc: allocate m times n bytes initialized to 0