2581-4575

International Journal For Recent Developments in Science & Technology

Crossref

A Peer Reviewed Research Journal

A STUDY OF DIFFERENT TYPES OF DATA STRUCTURES

1.A.SRAVANTHI, 2.Dr.U.M.FERNANDES DIMLO

[1]Assistant Professor, Department of CSE , Narsimha Reddy Engineering college

[2]Professor, Department of CSE ,Narsimha Reddy Engineering college

[1]sravanthi810@gmail.com, [2]umfernandesdimlo@gmail.com

ABSTRACT

There are many popular problems in different fields of computer science, Data Mining, Information Retrieval systems Networks and Artificial intelligence. The basic operation is of storing data in a specific format which is organized in an efficient manner to retrieve and the problem has attracted a great deal of research. This research paper presents the different types of data structures like Stack, Queue, Hash table, Tree, Graph. This paper provides a detailed idea of how these data structures work their advantage, disadvantage and applications.

1. INTRODUCTION

Data Structure can be defined as the group of data elements which provides an efficient way of storing and organizing data in the computer so that it can be used efficiently. Data structures are the basic building blocks of any program or the software. Choosing the proper data structure for a program is the most difficult task for a programmer. As applications are getting complexed and amount of data is increasing day by day, there may arise the following problems:

Processor speed: To handle very huge amount of data, high processor speed is required, but as the data is growing day by day to the billions of files per entity, processor may fail to deal with that much amount of data.

Data Search: Consider an inventory size of 200 elements in a store , If our application needs to search for a particular item, it needs to traverse 200 elements every time, results in slowing down the search process.

2. STACK

It's a collection of elements where we add and remove items to and from the top of the stack, insertion is called push and deletion is called pop A stack can be easily implemented either using an static array or a dynamic linked list. Stacks follows last in first out (LIFO).Stack is not limited to model only real world problems. One of the best uses for a stack in programming is when parsing code i.e syntax parsing or expressions, like validating expression evaluation which includes the correct amount of opening and closing curly braces, square brackets or parenthesis.

ADVANTAGES:

1. When function is called the local variables are stored in stack and destroyed once returned. Stack is used when variable is not used outside the function.
2. Stack gives control over how memory is allocated and deallocated for data.
3. Stack will cleanup(read delete) the object reduces the burden.
4. No one can easily insert data in middle because it uses LIFO

DISADVANTAGES:

1. Stack memory is limited.

2. Too many objects created on the stack will increase the chances of stack overflow

3. Random access not possible

APPLICATIONS OF STACK:

1. Recursion
2. Expression evaluations and conversions
3. Parsing
4. Browsers
5. Editors
6. Tree Traversals

3. QUEUE

It's a collection of items where we add items to the rear end and remove items from the front of the queue it is implemented either using an array or a linked list. Queues are usually used in concurrency situations to keep track tasks which are waiting to be performed and make definite we take them in that order.

ADVANTAGES:

1. Manages the data in particular way FIRST IN FIRST OUT (FIFO).

2. Not easily corrupted (No one can easily insert data in middle)

DISADVANTAGES:

1. Random access not possible

APPLICATIONS OF QUEUE:

1. Queues are used in managing waiting lists for a single shared resource like printer, disk, CPU.
2. Queues are used in asynchronous transfer of data (where data is not being transferred at the same rate between two processes) for eg. pipes, file IO, sockets.
3. Queues are used as buffers in most of the applications like MP3 media player, CD player, etc.
4. Queue are used to maintain the play list in media players in order to add and remove the songs from the play-list.
5. Queues are used in operating systems for handling interrupts.

4.TREES

A tree is a non linear data structure which stores hierarchical data a collection of nodes, and the nodes have connections, they have links between each other. It's a collection of nodes, where each node might link to one or more nodes.

4.1Binary Trees

A binary tree is just a tree with finite set of nodes maximum of two child nodes for any parent node. Binary trees are often used for implementing search structure called a "Binary Search Tree"

4.2Binary Search Trees (BST)

It's a specific type of binary tree, where the left sub tree contains all child node is less than its parent, and a right sub tree contains all child node is greater than its parent.

ADVANTAGES:

1. Searching an element is faster. We can search for a given key in moderate time using divide and conquer approach

2. Insertion and deletion of keys is done at moderate time

DISADVANTAGES:

1. High overhead

2. Large waste of unused links

3. Predetermined limit on number of a node's children

APPLICATIONS OF TREES

1. Store hierarchical data, like folder structure, organization structure, XML/HTML data.
2. Binary Search Tree is a tree that allows fast search, insert, delete on a sorted data. It also allows finding closest item
3. B-Tree and B+ Tree : They are used to implement indexing in databases.
4. Syntax Tree: Used in Compilers.
5. Trie : Used to implement dictionaries with prefix lookup.
6. Suffix Tree : For quick pattern searching in a fixed text.
7. Spanning Trees and shortest path trees are used in routers and bridges respectively in computer networks

5.HEAPS

Heaps are implemented using the basic idea of a binary tree, not a binary search tree but still, a binary tree. It is a way of implementing **priority queue**.It's a specific type of binary tree, where we add nodes from top to bottom, left to right, and child nodes must be less (or greater) than or equal their parents.

ADVANTAGES:

1. When the data in the variable is needed beyond the lifetime of the current function.

2. It provides the maximum memory an OS can provide

DISADVANTAGES:

1. You must make sure to free the memory of the data when you are done with all its operations.

APPLICATIONS OF HEAP

1. Heap is a tree data structure which is implemented using arrays and used to implement priority queues.
2. Heaps are used in many famous algorithms such as Dijkstra's algorithm for finding the shortest path
3. It is used in Heap sort

6.GRAPHS

A graph is a non linear data structures consists of nodes (vertices) and edges. A node can link to multiple other nodes, there is no specific sequence, no root node. Two adjacent vertices are joined by edges. Basically graphs are two types directed and undirected graph.

ADVANTAGES:

1. Finding the path in efficient way

DISADVANTAGES:

1. High overhead

2. Large waste of unused links

APPLICATIONS OF GRAPHS

1. Google maps uses graphs for building transportation systems,
2. In Facebook , users are considered to be the vertices and if they are friends then there is an edge running between them
3. In Operating System, we come across the Resource Allocation Graph where each process and resources are considered to be vertices. Edges are drawn from resources to the allocated process, or from requesting process to the requested resource.
4. Graphs in compilers. Graphs are used extensively in compilers. They can be used for type inference, for so called data flow analysis, register allocation and many other purposes.
5. They are also used in specialized compilers, such as query optimization in database management systems

7.CONCLUSION

Data structures are the basic building blocks that we use to organize all of our digital information. Choosing the exact data structure allows us to use the algorithms we want and keeps our code running smoothly. Understanding data structures how to use and implement them well can play a vital role in many situations

REFERENCES

1. http://www.mylearning.in/
2. www.coursehero.com
3. www.geeksforgeeks.org