

DATA STRUCTURES AND ALGORITHMS

STEP 1: REPEAT STEP 2 TO 4 FOR $i = 1$ TO N
 STEP 2: REPEAT STEP 3 TO F FOR $I = 1$ TO M
 STEP 3: SET $C[I][J] = 0$
 STEP 4: REPEAT FOR $K = 1$ TO P
 STEP 5: SET $C[i][j] = C[i][j] + A[i][K] * B[k][j]$
 STEP 6: EXIT





"LEARNING STARTS WITH VIEWING THE WORLD DIFFERENTLY."

Knowledge flow- A mobile learning platform provides Apps and Books.

Knowledge flow brings you learning book of **Data Structures and Algorithms**. This book is for all information technology and computer science students and professionals across the world.

Follow us on

[Facebook](#)

[Google plus](#)

[Twitter](#)

For more information visit us at

[Knowledgeflow.in](#)

[knowledgeflowapps.blogspot.in](#)

Thank you for using Knowledge flow eBooks



DATA STRUCTURES AND ALGORITHMS

1. Introduction
2. Data structure for string and pattern matching Algorithm
3. Arrays and Pointers
4. Linked Lists
5. Stacks and Queues
6. Trees
7. Graphs in Data Structure
8. Sort and Search
9. AVL Search Trees
10. Warshall's Algorithm and shortest path

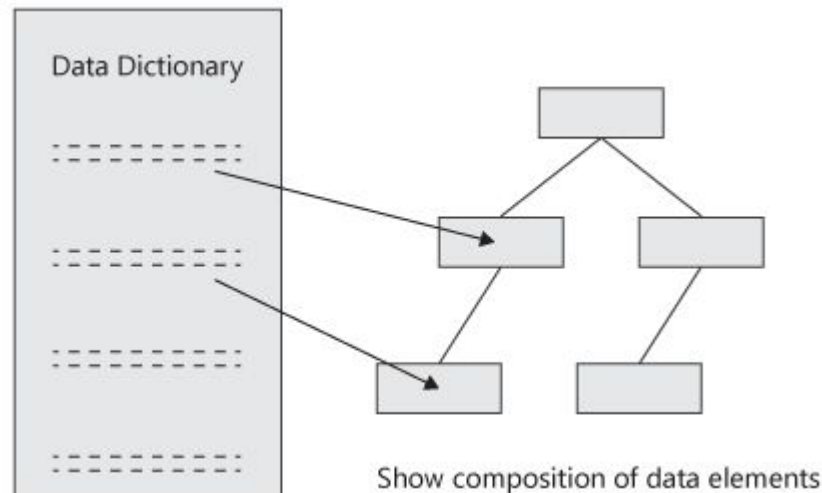


Disclaimer

This eBook contents is for informational and study purposes only. The **Knowledge flow** makes no claims, promises, or guarantees about the accuracy, completeness, or adequacy of the contents of this eBook and no legal liability or other responsibility is accepted by **Knowledge flow** for any errors, omissions, or statements on this eBook.



Data Structures is a study or a technique used to understand how to store a bunch of data in an organized manner, so it can be used in a very sophisticated manner for designing of programs and algorithms.



Data structure diagram

Data structure is a vital subject to the discipline of Information and Computer technology. In simple words Data structure is nothing but storing and then fetching of the data from any part of the memory of the computer in an organized manner whenever needed.

The term Data Structure was first coined into existence in year 1976 written by Niklaus Wirth in his book "Algorithms + Data structures = Programs", which covered some fundamental details of computer programming.

Types of data structures

- Arrays: Simplest data structure which shows linearity as a property. The elements are arranged in a linear manner. Let us choose X as a

name for the array. And the elements are referenced respectively by numbers 1,2,3,...,n .Therefore it can be denoted by

1. Subscript notation: $x_1, x_2, x_3, \dots, x_n$;
2. By parenthesis notation: $K(1), K(2), K(3), \dots, K(N)$;
3. By bracket notation: $K[1], K[2], K[3], \dots, K[N]$.

Now Y in $K[Y]$ is called subscript and $K[Y]$ in whole is called a subscripted variable.

- **Linked Lists:** In this type of data structure it's difficult to fetch data from the data since it might be massive. In this particular situation we use the tools like pointer and link. Both pointer and link have different functionality.
- **Pointer:** is used when it's needed for an element in one list to point to an element in a different list. Link is used when an element in a list points to an element in that same list.
- **Trees:** This type represents the whole data present in a hierarchical form on the basis of the relationship between the elements in that group of data.
- **Stack:** It's last in first out. In this type of data structure the data are inserted and deleted from one point. It's just one way in and out.
- **Queue:** It is first in first out. In this data is inserted from one side which will be the front and pushed to the other side till it reaches the end that is the rear side. It has two ways, one way in and another way out.
- **Graph:** It depicts relationship between pairs of elements.

Operations of data structures

- **Traversing:** Accessing or visiting the record to be processed.

- Searching: Finding the record or the location of the record with the key value.
- Insertion: Adding a record.
- Deletion: Removal of record.
- Sorting: Arranging records in the needed order.
- Merging: Combining two records into a single file.



Strings in data structure are character data in sequence.

Characters used are:

Alphabet: A, B, C,..., Z.

Digit: 0 1 2 9.

Special characters are: + - / * () , . \$ = '

Storing of strings

- String can be stored in three types of structures:
- Fixed-length structures,
- Variable-length structures with maximum limit set,
- Three linked structures.

Fixed-length storage

In this each line is seen as a record and all records have the same length.

Main advantages of this way of storing strings are:

- Easy to access data.
- Easy to update data.

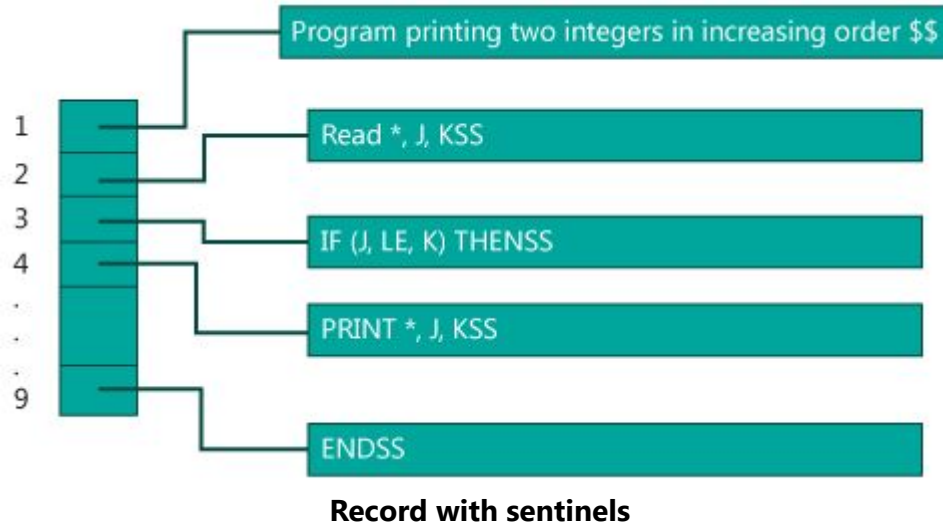
Main disadvantages of this kind of storing technique:

- Waste of time due to inessential blank space.
- Changing even a misspelled word leads to change in the whole record.

Variable-length structure with maximum limit set

The storage can be done in two general ways:

- Use of a marker to show the end of the string.
- Length of the string can be listed as an item in the pointer array.



Linked storage

In this memory cells are assigned to a character and also a link that would provide them the address of the cell that has the next character in the string.

End of this sample Kindle book.
Enjoyed the sample?

[Buy Now](#)

or

[See details for this book in the Kindle Store](#)
