

Data Structure

(1)

Data $\Rightarrow$ Anything to give information is called data.

Ex $\Rightarrow$ StudentName, Student Roll no.

Structure $\Rightarrow$ Representation of data is called structure.

Ex $\Rightarrow$ graph, Arrays, List.

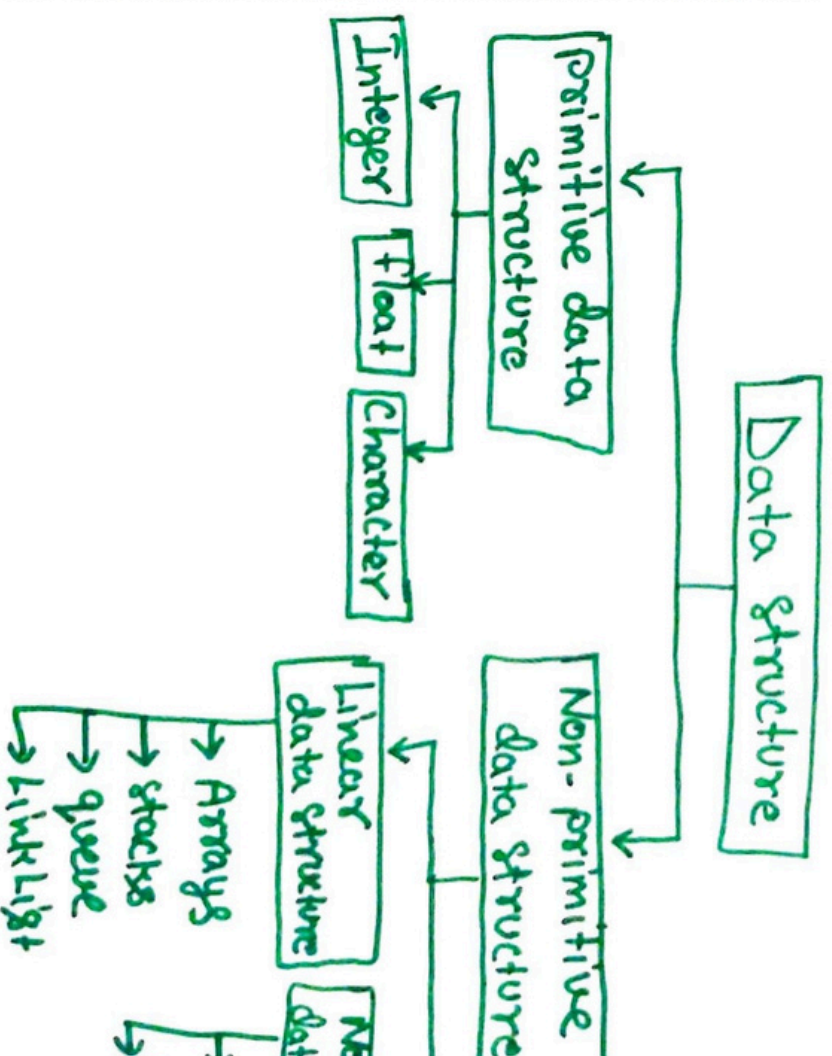
Data Structure $\Rightarrow$

- Data Structure = Data + Structure
- Data Structure is a way to store and organize data so that it can be used efficiently (better way)

- Data structure is a way of organizing all data items and relationship each other.

Types of data Structure $\Rightarrow$

There are mainly two types of data structure.



(3)

Primitive data structure $\Rightarrow$ These are basic structure and are directly operated by machine instruction.

Ex $\Rightarrow$ integer, float, character.

Non-primitive data structure $\Rightarrow$ These are derived from the primitive data structure. it's a collection of same type or different type primitive data structure.

Ex $\Rightarrow$ Arrays, stack, trees.

(4)

Data Structure operation $\Rightarrow$

The data which is stored in our data structure are processed by some set of operation

- i) Insertion $\Rightarrow$ Add a new data in the data structure
- ii) Deleting $\Rightarrow$ Remove a data from the data structure
- iii) Sorting $\Rightarrow$ Arrange data in increasing or decreasing order.
- iv) Searching $\Rightarrow$ find the location of data in data structure.

v) Merging $\Rightarrow$ Combining the data of two different sorted files into a single sorted file.

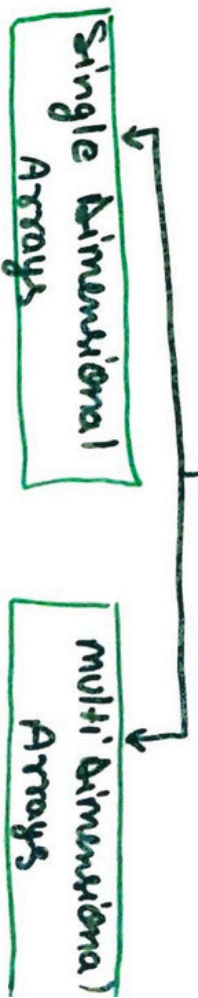
vi) Traversing $\Rightarrow$ Accessing each data exactly one in the data structure so that each data item is traversed or visited.

Arrays

(5)

- An Array can be defined as an infinite collection of homogeneous (similar type) elements.
- Array are always stored in consecutive (specific) memory location.
- Array can be store multiple values which can be referenced by a single name.

Types of Arrays



1) Single Dimensional Arrays $\Rightarrow$ It's also known as

One Dimensional (1D) Array.

- It's use only one subscript to define the elements of Arrays.

[row] [col]

Declaration $\Rightarrow$

(6)

data-type var-name [expression],

Ex $\Rightarrow$

int num[10];

char c[5];

Initializing one-dimensional Array $\Rightarrow$

Data-type var-name [expression] = {values};

Ex $\Rightarrow$

int num[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

char a[5] = {'a', 'b', 'c', 'd', 'e'};

2) Multi-dimensional Arrays $\Rightarrow$ multidimensional

Arrays use more

than one subscript to describe the Arrays elements. [][][] ---

Two Dimensional Arrays $\Rightarrow$ It's use two

[row] [col]

Subscript, one subscript

to represent row value and second subscript to represent column value.

It's mainly use for matrix Representation.

• Write a program to read & write one-dimensional array.

One-dimensional array.

• Include <stdio.h> - standard input and output stuff

• Include <conio.h> - console input and output

void main()

{ clrscr(); int a[10], i;

printf("Enter the array elements");

for (i=0; i<10; i++)

{ scanf("%d", &a[i]);

printf("The entered array is");

for (i=0; i<10; i++)

{ printf("%d\\n", a[i]);

}

printf("The entered array is");

for (i=0; i<10; i++)

{ printf("%d\\n", a[i]);

}

getch();

}

}

Stacks

(Data Structure)

• Stack is a linear structure.

• It is an order of already existing from only one top of stack.

• The last added first to be removed. This is the first-in-first-out (FIFO) principle.

• The last added first to be removed. This is the first-in-first-out (FIFO) principle.

• The last added first to be removed. This is the first-in-first-out (FIFO) principle.

• The last added first to be removed. This is the first-in-first-out (FIFO) principle.

• The last added first to be removed. This is the first-in-first-out (FIFO) principle.

• The last added first to be removed. This is the first-in-first-out (FIFO) principle.

• The last added first to be removed. This is the first-in-first-out (FIFO) principle.

• The last added first to be removed. This is the first-in-first-out (FIFO) principle.

Want full handwritten notes?

Unlock complete notes for just ₹_9



WhatsApp: +91-7037206791

⑥

Operations on Stack.

There are two operations of stack.

1) PUSH operation - The process of adding

a new element to the top of stack is

called PUSH operation.

- Every new element is added to stack top is incremented by one.

- In case the array is full and no new element can be added it is called Stack full or

Stack overflow condition.

2) POP operation - The process of

deleting an element from the

top of stack is called POP operation.

- After every pop operation the

Stack is decremented by one.

- If there is no element on the stack and the pop is performed then this will

result into Stack underflow condition.

Stack operations

- Stack has two

1) PUSH operation

2) POP operation

1) PUSH operation

of the top of stack operation

- Every push operation

by one.

Top

- In case the array element is added

Called Stack full

Condition.

Want full handwritten notes?

Unlock complete notes for just ₹_9



WhatsApp: +91-7037206791

Algorithm for inserting an item into the stack (push operation).

push (stack[main], item)

Step 1: initialize

Set top = -1

Step 2: Repeat steps 3 to 5 until Top < main-1

Step 3: Read Item

Step 4: Set top = top + 1

Step 5: Set stack[top] = item

Step 6: Print "Stack overflow"

2. pop operation

- The process of removing an item from the top of the stack is called pop operation.

- After every pop operation,

Stack Top is

Top = Top - 1

- If there is no

Stack and the

is performed +

result into STA

Condition.

Stack

Top = Top - 1
= 2 - 1
= 1

Want full handwritten notes?

Unlock complete notes for just ₹_9

WhatsApp: +91-7037206791