Chap : 1

Concept of data structure

30th Nov

Data structure

↳ It is a particular way of storing & organizing data in a computer so that it can be used efficiently.

- Diff. ways to organize data:
  - i) stack
  - ii) queue
  - iii) List
  - iv) Tree
  - v) Graph & so on

classification of data structure

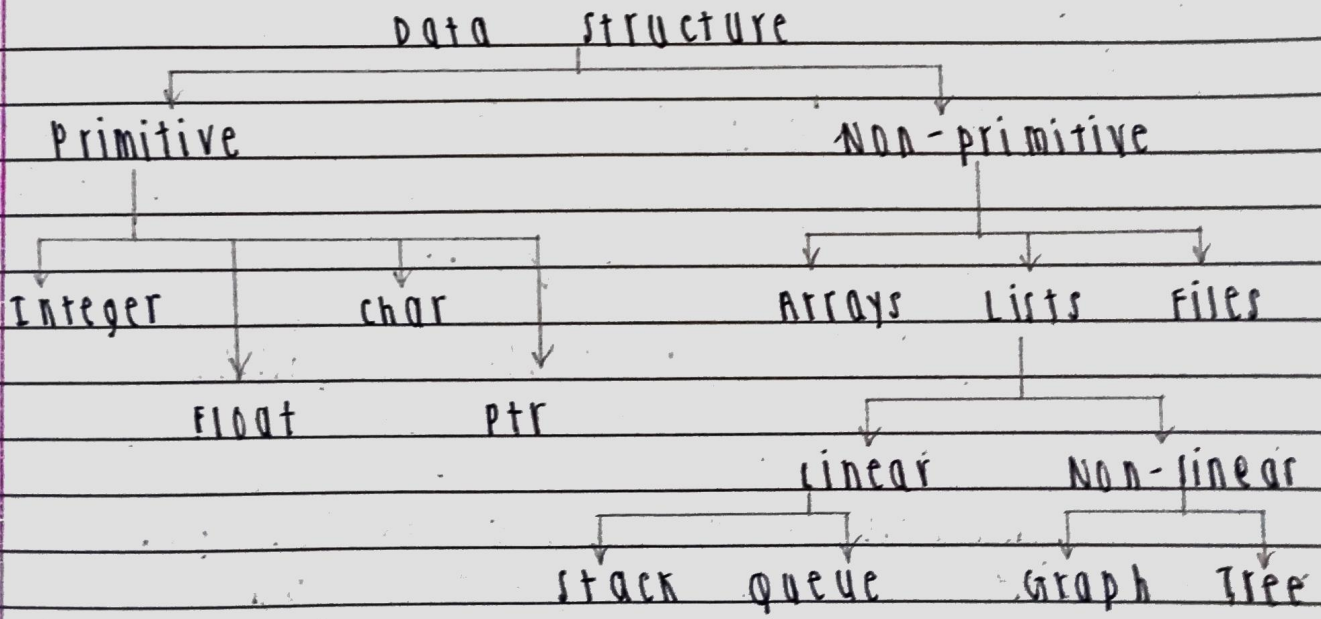

Fig: classification of data structure

Note: • stack & queue → restricted

& &

LIFO FIFO

(last In 1st out) (1st In 1st out)

• Array → unrestricted

&

jata ni store, delete garna milxa

Abstract Data Type (ADT)

↳ ADT is a basic mathematical concept that defines the data type.

↳ ADT focuses on what it does

&

ignores how it does its job.

↳ ADT consists of two parts

• value definition

& • operator definition

• Illustration

To illustrate the concept of an ADT, consider the ADT RATIONAL which corresponds to the mathematical concept of a rational no.

/* value definition */

abstract typedef <integer, integer> RATIONAL;
condition RATIONAL [1] != 0;

/* operator definition */

abstract RATIONAL makerational (a,b)
int a,b;

pre condition b != 0;

post condition makerational [0] = a;
makerational [1] = b;

abstract RATIONAL add (a,b) // add parse fn
RATIONAL a,b;

post condition add [1] == a[1] x b[1];
add [0] == a[0] x b[1] + b[0] x a[1];

add parei x hunxa vanera dekauxa

> $\frac{1}{2} = \frac{2}{4}$ → 1 x 1 = 2 x 2

abstract RATIONAL equal (a, b) // fn for compare
RATIONAL a, b;

post condition equal == (a[0] x b[1] == b[0] x a[1])

Algorithm -- Defn
↳ time, space complexity

Date ______
Page ______

chap : 2
The stack & Queue

stack

↳ It is an ordered collection of items into which items may be inserted & from which items may be deleted from one end which is known as top of stack.

↳ stack is Last In First Out (LIFO) type data structure.

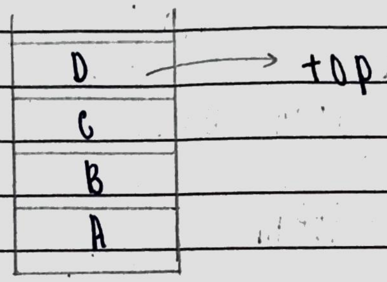

Fig: stack

• operations on stack
① Push
② Pop
③ Display operations

Algorithm

① Declare & initialize necessary variables:
- top = -1
- max size
- item
- stack [max size]

② For push operation

If top == (max size - 1)
print "stack is full"

else
top = top + 1
Read item from user

stack [top] = item

③ For next push operation, repeat step 2.

④ For pop operation

If (top == -1)
print "stack is empty"

else
item = stack [top]
top = top - 1

display item