Time complenity

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

Prime -> 2 factors

others

→ for i → 2 to n-1 if divides by n it's not prine no.

-9 11 - 2 = 9ms

 $\Rightarrow n = .10^6 + 3 = 10^6 \text{ ms}$ $\approx 10^6 (\text{s.c.}) \Rightarrow 16.66 \text{ mind}$ $10^{10} + 19 = 10^{10} \text{ ms}$ =) $10^{10} \text{ J.c.} =) 115 \text{ days}$. for etaln if e divide by
n it is not prime

VII → 3-1 = 2 ms

> VIOI > 9 ms

V166 +3-1 = 103 ms

= 1sec

V1610 +3-1=103 ms=1 sec

insertion > tree retraining - hashing

24/11/2016

DATA STRUCTURES

Defination: -

1 "The logical and mathematical representation of data in the computer memory is called as data-structure".

I "A lata structure is a method of storing data in a computer so that it can be used refficiently!

=> The data structures mainly deals with the study of how the data is organised in the memory.

of How efficiently the date can be stored in the memory.

>> How efficiently the data can be retrived and manupulated

-) Data Structures it is telle you the representation of logical relationships between elements of data

79n other words a date structure is a way of organising data items by considering its relationship to each other.

The Data structures affects the design of both structural and functional aspects of a program.

Algorithm of data Structure = Program!

The representation of particular clute stoucture is the memory of a computer is called as storage stoucture is e.e. a data structure should be representated in such a way i.e utilized manimum esticiency.

The data structures can be maintained in both main memory and (seconf)aunillary memory of the computer.

- offe often known as "file structure".
- I finally that the date structures and the operations on organized data items can combinly solving the problem using computer.

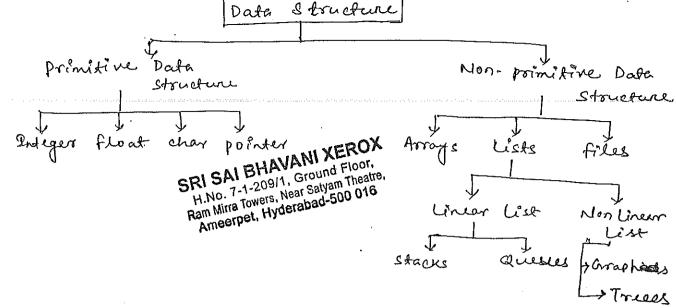
Data Structure = Organized Data + Operations

- → The desta structures are classified in two ways

 1. Primitive Data etructures

 2. Non primitive data etructures.
- The pointive data stouctures are the data stouctures that can be manupulated directly by the machine statements.
- I Non prinitive data structures cannot be manupulated directly by the machine statements. These non-pointines are of two types a. Linear

b. Non-Linear.



) In many areas data structures are mostly implemented like: compiler Design, Operating Systems, Data Management - Systems, statistical Analysis package, Numerical - Analysis, complices, Artificial Intelligence, Simulation

> The major Data Stouetures using the following areas RDBMS, Network Data Model and Hierarchial Data Model

7 RDBMS -> Array (i.e Array of Structures)
7 Network Data Model -> Trees.

25/11/2016

ARRAYS

As we know that primitive data structures like integer, float etc, they can store only small amount at data, we need to declare number of variables, for suppose, of want to stone manks in three subjects is need to declare 3 variables, for or subjects, or memory location required, petining the memory locations is possible but it is not recomendable, since the chout and output statements and logical statements are increasing. To overcome this workship we use

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

non-primitive data structures.

Defination of Array

An array is an single subscripted Variable which can hold. In number of values in the single variable

Enample for static memory allocation is Arrays

Juhan we defining static memory allocations, the sixe of the array has to define, out the time of compilation. I without defining the size it is next possible to declare static memory. Allow

Synton:

datatype Var_name [size];

Enample: Ent arr[10];

ar constant base address

600 0 1 2 3 4 5 6 7 8 9

-600 602 04 06 08 10 12 14 16 18

Address of (0 r)

each block Subscript

(

- -7 cace we define the size of an array, you cannot concrease or decrease the size.
- -> An Array is an homogeneous type. (same type)
- -> An array elements will be identified logically by the Subscript numbers and phisically identified by the addresses
- The array subscript always starts with 'xero', for suppose 'n' is a size, then there will be (n-1) Subscript numbers are emisted.

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

```
a) write a program
                      accept
                               on array of elements and.
 display that.
                                        l'= inden No.
     main ()
                                       alight Elements of the array of Constant Base address
      isht a [100].
       int nit;
       claser():
      'printf (" Enter the size of n: "):
       'scanf (" /-d", &n);
      · for( (=0; (<0; i++)
          } prontf (" Enter the element [4-d]: ", i);
           scant (" y.d", 8 Tij);
        for (1, = 0) ( (0)
       3 green ( Ranning por
  write a program accept an element, delete that
  element from an array.
    merin()
    Ent atisoj,
      ent n, i, de, status = 0, j;
      clrscr();
      Prints (" Enter the size of n:");
      scomt ("">d ", &n);
     for (120; i(n; i++)
        Prints ("Enter the element at a [4d]: ", ");
         Scanf (" Y.d", 2 a [ i]);
```

```
Printf ("In Enter the deleting element
scanf ( " /d", a de);
Printf (" \n Before
                      deletion the elements
for (120; ((n; i+4)
print ("Y-3d", aris);
 for (120; i(n; i++)
         Status = 1 º
          for (j=1; j(n; j++)
             1+1] = a []+1].
              i -- ; // for duplicates
      3
   ?f (status == 0)
     Printf ("In Element not found");
    elec
   2 prints ("In Elements
                                    deletion. . In ").
      for (izo; i(n; i++)
         Printf ("7.3d", a [i]);
   geten ();
               SRI SAI BHAVANI XEROX
                H.No. 7-1-209/1, Ground Floor,
                Ram Mirra Towers, Near Salyam Theatre,
                 Ameerpet, Hyderabad-500 016
```

Disadrantage:

- -> As an array is an homogeneous it can hold, only same type of elements.
- of data.
- -) As an array is an efatic memory allocation the size has to define at the time of compilation. Once we define the size it is never possible to increase or decrease the size.
- Oncept.

SRI SAI BHAVANI XEROX

STACKS

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

- → A stack is an special data structure where elements are inserted from one end and elements are deleted from the same end.
- The position from where the elements are inverted and from where the elements are deleted is called as Top of the Stack
- >> Stark is also called as last in fact out data structure.
- > stack application can be implemented in two ways
 - 1. Arrays 2. Linked List.
- The general operations which takes place on stack
 - (e) Push -> Prierting an element on the top of the stack.
 - (ci) Pop -> Deleting an element from top of the stack.
 - (iii) Display -> Display the contexts of the stack.

-> whenh implementing by the linked list concept the no of operations can be changes.

I As we working with arrays concept we need to define the size of the array at the time of compilation

MAX = 5 size

→ initially stack is empty; it indecates underflow condition.

i.e there is no elements in the stack.

Stack.

At the time of push operation everytime

it checks the stack is overflow or not. if it is overflow

we cannot implement push operation.

of push operation the top will be incremented.

and then push operation takes place where ever the top
is pointing.

Thert 10

if (top == MAX-1) // overflow condition

" overflow"

if this condition not st satisfied we can implement puch operation by incrementing the top.

inden < top ++; 11 -1++=0

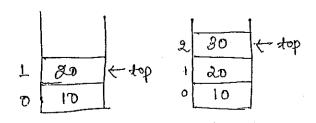
Stack [top] = ele;

arroy

SRI SAI BHAV ANI XEROX

Ground Floor,
Ground

=> Insert an elembents 20 & 30



ortish
Underflow
(no element)

oro (

C

```
pop operations, then it check the
           complement
  top
       condition
           if (t op ==-1)
               " Underflow"
            there is no elements on the stack. (empty stack)
              top operation takes place, where ever the top is
          that element logically delated in arrays by
  decrementing the top position.
                   deleted, thysically
=) It is logically
                                                   30
                                                          - stop
                                               1
                                                   20
    enisted.
                                                   10
-> Insert 40 in the stack
                                            <- t0₽
                                    1
                                       20
28/11/2016
                                         SRI SAI BHAVANI XEROX
  # define MAX 4
                                          H.No. 7-1-209/1, Ground Floor,
    ent stack [MAX], top=-1, ele;
                                          Ram Mirra Towers, Near Salyam Theatre,
                                           Ameerpet, Hyderabad-500 016
   Void push ();
    Void POP();
    Void Print ():
   main().
     Ent ch;
      Closer();
       do
        printf ("In 1. push").
         printf ("In a pop");
         printf ("\n 3. print").
         printf ("\n 4. Enit").
         printf ("In Enter the choice ").
         scanf ("/d", gch);
```

Switch (ch)

Case 1: push (); break;

Case 2 ; pope); break;

```
Case 3: print(); break.
  case 4: enit(0);
  while (ch ! = 4);
   getch();
void pund
  of (top = = MAX-1)
   print ("In overflow"):
   else
  } printf ("Enter the element 6").
    scanf ("yd", qele);
     top++;
    Stack [top] = ele:
                                7.1.2091, Ground
                              an wing toners hear sayam hears
  Void pope)
   (f (top = = -1)
       Printf ("underflow \n");
       ele = stack[top];
         top--
         Printf ("In the deleted element is: 1.d", ele);
   Void
         print ()
      if (top == -1)
     else
      { for ( i= top ; i >= 0; i--)
        Print ("%d", & stack[i]);
```

=> Based on the stack decipline in Last on fast out, number of applications like, balancing of symbols, conversion of enpressions (Infix to postfix etc),

> Evaluations of the enpression.

- Implementing function calls (Indebuding Reconsion)

> Undo sequences int a tent editor (chol +2)

> page-visited History in a web browser (Back bottom)

-finding of spans (stock market)

-> Matching Tags in HTML and XML etc.

Conversions and Evaluation of Expressions:

* what is an enpression?

An enpression is a combination of sequence of operators that reduces to a single value after. and openands evaluation is called as an enpression.

This enpressions are of 3 types

1. Infin enpression

2. prefix enpression

3. postfin enpression.

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

1 . Infin Enpression:-

a enpression i't an operator is placed in between the operands, the enpression is called infin enpression.

Example: (A+(B-C))

20 prefix expression:

In an enpression of an operators state splaced before the operands such type of enpression scalled as prefix en marrion.

Example: +a-bc tax bc

3. post fix expression :-

In an enpression operands let and then operators such type of enpression is called as post-fin enpression.

Enample: abc-+
abc+

Note: - Always the compilers try to convert the enpression (into the post-fix enpression.

=> Infin to prefix conversion: (a+b) * (a-c)

The below table shows anotheratic operators along with priority values and their associativity.

		+ 0	
Du cription	operator	7 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Associativity
Enponentiation Multiplication		TO SER 4	Right to Left Left to Right
Division	*/*	Le Co	L to R
Addition Substraction	_	RI SAI BHAVANI X	L to R (EROX to R

(a+b) * (a-c)

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

tab # - ac [& & y are operande now]

并 tab-ac

When

Note:-

- 10 whenever an enpression és converting, the enpression after conversion treated as an operand
- the time of conversion do not change the order of the openands

29/11/2016

$$(A+(B-c))$$

$$(A + * - BCD)$$

$$+ A * - BCD$$

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

$$((A+BC-*D)AE+F)$$

ABC-D*+EAF+

Algorithm 1 ?-

Infin to post-fin Conversion:

scan from	L to R Repeat step 1-4
Token	openation peration
operand	Add to enpression this section
C	Push to stack the Holliston
openados	pop operation, if P(popped)>=P(scanned) add to enpression. Push scanned operator to stack
	pop till (Add popped tocken to enpression, Delete (
h -	

pop stak elements it any add to enpression

A+(B*C-(0/E\$F)*G)*H

Token	Stack	Enpression	
# A +	# # # # # # # # # # # # # # # # # # #	4 6 4	Popped -) top of stack Scanned - new
C B	#+ #+ #+ #+ #+ #+ #+ #+ #+ #+	A AB	# Scanned
* C -	#+(* * * * * * * * * * * * * * * * * * *	ABC PABC*	# > = + opped 0 > = 2 x
(D	#+(-(#+(-(ABC *	+ (>=* 0>=4×
/	#+(-()	ABC*D	*>=- 4>=2~
E	#+(-(/	ABC * DE	6>=/ 0>=4 X

\$ # +(-(/\$ ABC * DE f #+(-(/\$ ABC * DEF #+(-ABC * DEF \$/ ABC * DEF \$ / * #+(-* RISAIBANAMI XEROS. * DEF \$/G

RISAIBANAMI XEROS. * DEF \$/G

HINO: 1-1-20911. Wear Salu-500 DEF \$/G * —

RISAINITE TOWNERS. HVderabad-5-4BC * DEF \$/G * —

RISAINITE TOWNERS. HVderabad-5-4BC * ABC * A #+(-* * ABC & DEF \$ 1 G *- H H #+* ABC * DEF \$ / G * - H*+

A+(B*C-(D/E\$F)*G)*H

A+(B*C-(D/EF\$)*G)*H

A+(B*C-(D/EF\$)*G)*H

A+(B*C-(D/EF\$)*G)*H

A+(B*C-(D/EF\$)*G)*H

A+(B*C*-(D/EF\$)*G)*H

A+(B*

SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, H.No. 7-1-209/1, Ground Theatre, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

SRI SAI BHAVANI XEROX
SRI SAI BHAVANI XEROX
H.No. 7-1-209/1, Ground Floor,
Theatre,
H.No. 7-1-209/1, Ground Floor,
H.No. 7-1-209/1, Ground Floor,
H.No. 7-1-209/1, Ground Floor,
H.No. 7-1-209/1, Ground Floor,
Ram Mirra Towers, Near Satyam On 6
Ram Mirra Towers, Near Satyam On 6
Ameerpet, Hyderabad-500 016

Token	Stack	Enpression
#	#	4
A	#	A
. \$	## 4	# 7 = 4
В	#\$ 20	AB \$>=*
*	# * # * # * # * # * # * # * # * # * # *	# 3 × 3 = -
C	# * 11/1/20 20 20 20 20 20 20 20 20 20 20 20 20 2	AB & C
****	# - 67 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8	AB\$C*
D	# - 5 1. 10 10 10 1	AB\$C*D
+	#+	AB\$ C * D-
E	#+	AB\$C*D-E
/	# +/	AB\$C*D-E
F	#+/	AB & C * D-EF
• • • • • • • • • • • • • • • • • • • •	#+/	AB\$C*D-EF/
(#+10	AB\$C* D-EF/
9	#+/(AB&C#D-EF/G
+	#+10+	AB\$C*D-EF/G
H _	# +1(+	AB\$ C*D-BF/94
	# +1C+ # +1	AB\$C*D-EF/GH+
	VERON	AB\$C * D- EF/ GH+/+

SRI SAI BHAVANI XERON SRI SAI BHAVANI XERON Ground Theatre. Ground Theatre. H.No. 7-1-20911, Ground Theatre. H.No. 7-1-20911, Ground Theatre. Ram Mirra Towers, Near Sadad-500 046. Ram Mirra Towers, Hyderabad-500 046. Ram Mirra Towers, Hyderabad-500 046.

```
30/11/2016
               on Infin to pulfix (suffin) conversion.
 // program
 #include (stdio.h)
 # include (ctype h)
# define MAX 20
                                            D; ele, ch, x, t;
  Char Infin[MAX], POST[MAX], S. [4]
  int i=0, top =-4, j=0;
  void push (char ch)
    top ++;
    STtop] = ch;
   Char pop()
    { ele = s[top];
                          return (s[+op]-- ]).
       top --;
     return ele;
   int priority (char ch)
     } if (ch == 'n')
            return 4:
        else if (ch == (*) || ch = 1/1 || ch == 1%)
            return 3;
         else if (ch == '+'|| ch == '-')
                                 SRI SAI BHAVANI XEROX
             return 2.
                                  H.No. 7-1-209/1, Ground Floor,
                                  Ram Mirra Towers, Near Satyam Theatre,
                                   Ameerpet, Hyderabad-500 016
              neturn 0;
   Void check ()
        while (priority (+) <= priority (s(top]))
          Past [j++] = pop().
```

```
maine
   chrscre).
  Printf ("Enter the Infin Enpression: ").
  scand ("%s", infin),
                                                SRISAI BHAVANI XEROX
   push (+#1);
                                                   H.NO. 7-1-20911, Ground Thomas
English France More Common Thomas
                                                   H.No. 7-1-20911, Ground Floor, Theatre, No. 7-1-20911, Ground Floor, Theatre, Near Salyam Theatre, Near Salyam Theatre, Near Salyam O16
Ram Mira Towars, Near Salyam O16
Ram Mira Towars, Hyderabad-500 016
Ameerpet, Hyderabad-500
  while (infin [i] != (10')
    it=chfin[i];
        if (isalpha (+))
        post[j++] =+;
       else
           if (t=='+'||t=='-'||t=='*, ||t=='/'||*=='\")
                                                     || t = = '/' || t = = '(' || t = = ')')
               Switch (t)
                                      & push (t);
                                         H.NO. 7- Towner Moor Column Theatr
                  Case 1+1
                                         H.NO. 7-1-209/1, Ground Fround the Ram Mirra Towers, Near Sayam to 016
                                            am nima lowers, near salvam inearte
Ameerpet, Hyderebad-500 of te
Ameerpet, Hyderebad-500
                  Case 1 -1:
                 Carle (%)
                 Case 1 /1
                 Case (*1)
                 Case 'N':
                                        Check ();
                                        push (*);
                                         break.
                                        X = pop();
                                        Post[j++] = x;
                                   while (x!= (');
                                    break;
```

i=i+1;

} // while

while (sltopj!= '#')

{

post[j++] = pop();

}

post[j]='\o';

Printf("The postfin Notation of given infin is %s", past);

getch();

} // main

O/P-> Enter the Infin Enpression: (a+b)*(a-c)
The postfin Nobation of given infin is ab+ac-*.

1/12/2016

Evaluation of postfin Enpression :>

As the compiler make a conversion into the postfin it is not necessary to give the priorities to the operators. Postfix has to evaluation based on the algorithm.

→ Scan from L to R Repeat Step 1 - 2.

Token	Add to chack SRISAT 20911. Ground Floatien H.NO. 7 TOWERS NEED SAVAM TO 018
operand	Add to stack SRISAI BHAVAIN X HOOF. Add to stack SRISAI BHAVAIN X HOOF. H.NO. 7-1-20911. Ground Thealten H.NO. 7-1-20911. Ground Thealten
openator	pop stack ento n 1 Ram Ameerpet.
	perform n3 = na operator ns (order)

pop stack to obtain the nexult.

42\$3#3-84/11+/+

Token	Stack
4	4
2	4,2
\$	16 N3 = N2 openator N
3	16 13 18 18 18 18 18 18 18 18 18
*	16,3 48 48.3 SRI SAI BILAVANI XEROOTIO 48.3 SRI SAI BILAVANI XEROOTIO 48.3 SRI SAI BILAVANI ATEROOTIO 7,24, 72=8 73=72/71
3	48,3 GRINO. Towers bed sales of the north of
- 8	Rainas
4	
,	45, 8,4
1	45,2
<u>1</u>	45,2,1
<u>+</u> +	45,2,1,1
T.,	45,2,2.
	45, 2, 2, 1. 45, 2, 2, 1. 45, 1. 46 SRI SAI BHAVANI XEROX SRI SAI TONOIS TOWNER TO BE TO DE
+	4.6 SRINO TO TOWN HYDER

Token	Stack	
A	6	
В	6,3	
C	6,3,2	
-	\$, 1	
D	6,1,5	
- Xi -	615	
+.	11	SRI SAI BHAVANI XEROX
E	11,1	SRI SAI BHAVAN H.No. 7-1-209/1, Ground Floor, H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ram Mirra Towers, Hyderabad-500 016
\$	11	Ram Mirra Towers, Near Satyani 11016 Ram Mirra Towers, Near Satyani 11016 Ameerpet, Hyderabad-500 016
F	11,7	
+	18	
Program on pe	st fix evalu	cation.
# include < std # include < ct # include < ct # include < ma # define MAX 5	pe·h> th·h>	
chan suf[MAX], eint i=0,j=0 float pop() { return (SE } Void push (SE fop+t; S[top]=V	op 1, op 2, op, 1 top = -1;	Al, temp, nes; Al, temp, nes;

```
float operate (float ops, float ops, charch)
   } switch (ch)
      Z
        Case (+1: temp = 0P1 + OP2;
                   break,
        case ! _ ! "
                   temp = OPI - OPQ;
        Call 1 of 1: lemp = OPI * OP2;
                     bruen.
         case 1/1: temp = OPI - OPa;
                      break;
         Case ' N' : temp = spow (off , (int) ofa);
                      break:
        return temp;
  main ()
   ર્ટ્
     Urscre).
     Printf (" Enter the systim
      scanf (" y. s", suf)
      While ( Suffij
          ch = Suf Ex
          14 ( 15 3 3 3 ( Ch))
              printf (" Enter the value for 1.c=", ch).
              Scanf (" y. + ", &val).
             ? Push (val);
           else
           日(ch==(+)1|ch==(-)1|ch==(*)11ch==(*)11
                               Ch = = 1 % 11 ch = = (N).
```

```
OP2 = POP();
      OPL = POP().
      res = operate (op, op, ch);
       push (res).
                                        SRI SAI BHAVANI XEROX
                                         H.No. 7-1-209/1, Ground Floor,
                                        Ram Mirra Towers, Near Satyam Theatre,
  i= i+1;
                                         Ameerpet, Hyderabad-500 016
  } // wehile
 temp = pop().
   print (" The simplified answer for 1/2 = ", suf) ,
   printf (" % + ", temp);
   geten();
  } / main
9P>
  Enter the suffix enpression: ABC #- D*+ERF+
  Enter the value for A = 6
  Enter the Value for B-3
```

The simplified answer for ABC-DX+ENF+ = 18.000000.

F = 7

SRI SAI BHAVANI XEROX

SRI SAI BHAVANI XEROX

Ground Floor,

H.No. 7-1-20911, Ground Floor,

H.No. 7-1-20911, Near Salyam Theatre,

H.No. 7-1-20911, Ground Floor,

Ram Mirra Towers, Near Salyam Theatre,

Ram Mirra Towers, Near

Algorithm -3:

Infin to prefix conversion.

Scon'from . R to L nepeat step 1-4

Token	operation
operand	Add to Empression
>.	of the police to stack
operator Bis	property of stack of popped) > P(scanned) of popped) > P(scanned) of popped) > P(scanned) of popped) > P(scanned)
C REPLACE	Pop till). Add popped token to enpression. Delete)
pop stack elem	

$\frac{1/12/2016}{A + (B*C - (D/E$F)*G)*H}$ R > L

	•	
Token	Stack	Empression EROX
#	#	MANI X Eloor
H	#	H SRISAI BHAVANI XEROX SRISAI BHAVANI XEROX SRISAI BHAVANI XEROX SRISAI BHAVANI XEROX H.NO. 7-1-20911, Ground Floor, Theatre, Near Salvam Tooler, Near Too
*	#*	H SRISAI BHAVANI XERO Floor, Ground Floor, Ground Floor, Ground Floor, Ground Floor, Ground Floor, Floor, Ground Floor, Floor, H.No. 7-1-20911, Ground Floor, H.No. 7-1-20911, Ground Floor, H.No. 7-1-20911, Ground Floor, H.No. 7-1-20911, Hyder Hyder Ameerpet, Hyder Ameerpet, Ameerpet, Y
)	#*)	H Ameer 0 > 4
e	#*>	HG > GH
*	#*)*	HG GIH
)	#*)*)	HG GH
-p	# *)*)	HOF FAH \$7/
\$	#*>*>\$	HOE FOH
E	#*)*)\$	HULE BEUH
	#*)*>/	HOFE \$ 4 EFGH
•		

| (*(*# HOFE&D D HOFE \$ D/ # *)* HOFE \$ D/* #*). HGFB\$D/*C #*)-HGFES D/XC * * - (* # HGFE\$D/*CB # #) - # RISAIBHAVANO THEORY GFE PD/# CB# - *

RISAIBHAVANO THEORY GFE PD/# CB# - *

H.NO. 7-1-20911, Near Salvam of Salvam o HOFE&D/XCBX-# * # + SRI SAI BHAVANI XERO #+ + Aweather, Hyderapag-200 A HUFE& D/*CB*-*A # HGFE\$D/* CB* -* A + TAX-*BC*/D\$EFGH

A+(B*C-(D/E\$F)*G)*H

A+(B*C-(D/\$EF)*G)*H

A+(B*C-(D/\$EF)*G)*H

A+(B*C-/D\$EF*G)*H

A+(B*C-/D\$EFG)*H

A+(B*C-/D\$EFG)*H

A+(B*C-/D\$EFG)*H

A+(B*C-/D\$EFG)*H

A+(B*C-/D\$EFG)*H

A+(B*C-/D\$EFG)*H

SRISAIBHAVAAII XEROX SRISAIBHAVAAII XEROX Thear Satvam Thears The Towers Hyderabad 500 or s Rem Mire Spet. Hyderabad 500 or s Rem Mire Spet. Hyderabad 500 or s

At the time of implementing the program after accepting the input of infin enpression 1st neverse the string and implement the program at time of displaying the output. once again neverse and display.

Algorithm - 4:-

Evaluate prefix

Scan from R to L repeat step 1-2

Token	operation with	de Lingle
operand	Add to stack Brook wells	ad for
openator	operation Add to stack Bittonia Con Pop stack Bittonia Con Pop stack Bittonia Con Pop stack Confidence Con Pop stack Confidence Con Pop stack Confidence Con Perform n3=n1 open	
	pop stack intom?	
	perform n3=n1 open	stor 12

pop stack to obtain nesult

+-* \$4233/184+11

Token	Stack	
1	1	M1 2 1
1	1,1	
+	2	n3=n1/n2
4	2,4	, , , ,
8-	2, 4,8	
/	2,2	
/	2, 4, 8 2, 2 1 SRISAI BHAVANI XEROX 1 1, 3 H.No. 7-1-20911, Ground Floor, Theatre, Near Salvam Theatre, Near Salvam Theatre, Near Salvam Hyderabad. 500 016 Ram Mira Towers, Near Salvam Ameerpet, Hyderabad. 500 016	
3	1 / 3 H.No. 7 Towers, the sale	
3	1, 3, 3	
ಸ	1,3,3,2	
4	1, 3, 3, g, y	
\$	1,3, 16 3,16	
*	1348	
_ 	45 1,45	·
	46	

Algorithm - 5

post to pres algorithm :-

scan from L to R Repeat 1-2

TOKEN

operation

operand

push to stack

SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Rem Mirra Tourse, Naca Satura Til

operator

propostack into S1

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

pop stack into s2

Concatenate operator sa si,

push the result on stack

pop stack to obtain the result.

4B\$C*D-EF/9H+/+

Token	stack	
A	A	
		£2 = 1
₿	A,B	\$ A B
\$	\$AB 51	\$ AB
C	\$AB,C	-, 110
*	*\$ ABC:	
\circ	* \$ ABC,D SRISAI BHAV	Carding Piuui
<u> </u>	H.No. 7-1-2097. Ram Mirra Towers, Ne Ameerpet, Hyder	וציייטוון וווסעווקו
E	- * \$ABCDIE	•
F	- * \$ ABC D.E, F	
/	\$-* \$ ABCD//EF	
,	-* \$ ABCD, IEF, G	
Ġ	-* \$ ABCD/EF,G,H	
H	-* & ABCD, /EF, +GH	
+	-* \$ ABCD, 11 EF+ 6H	
/	+- X \$ AB CD 11 EF + G. H	
+	14 A 1 3 - 1 1 / 1 1 1 / 1 GH	
•	170 000 1/11 T/10H	

Token	Enpression
4	A-
B	A,B
C	A, B, C
*/-	A, XBC
D	A, 米BC, D SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, H.No. 7-1-209/1, Ground Floor,
E	H.No. 7-1-2097; Ground Theatre, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016
F	A, *BC, D, E, F
\$	A, *BC, D, \$ EF
/	A, *BC,/D\$EF
9	A, *BC, 1D\$ EF, G
4	A, *BC, */D1EFG
	A, - * BC * /D\$ EFG
· 	A, -*BE*/D\$ EFG;H
*	A, *-*BC*/D\$EFGH
+	+ A*-*BC*/D\$EFGH
•	ERON

SAI EHAVANI XERONA Ground in Theate Ground in Theate Ground in Theate Ground in Theate HAN I TOWE Near Salten TOO OIL HAN MITE TOWE HAD A GOOD OIL HAN MITE TOWE HAD A GOOD OIL

```
Conversion of Enfix to prefix.
    # include ( sedio. h)
   # includes conio.h)
  # define MAX 50
 # enclude (string.h)
   chan SIMAXI, infinimaxI, pre [MAXI, ele, ch, x, t;
       int = 0, j=0, top = -1;
                                                                                                                 GRISAI BHAVANI XEROX
      void push ( ) char ch)
                                                                                                                        H.No. 71.20911. Ground Floor.
                                                                                                                       Ball Mills Towers, Mest Salvam Lings, Mest Park The State Under Control Contro
                { 8[++ top] = ch;
                                                                                                                             am wills lowers, west salvam uneante
Hyderabad-500 016
          char pope)
                          return (s[+op--));
        ent preority ( char ch)
                 ¿t (ch == (1))
                                       return 4;
                            if ( ch == (*) || ch == (/) || ch == 1/6)
                                          return 3:
                              if (ch == (+) 11 ch == (-1)
                                           return a;
                                @ else
                                                        return o;
Void check ()
                                uehile (priority (t) ( pri
                                      prepare [j++] = pop();
```

```
main()
            chrsere).
            Printf (" Enter the infin enpression ");
            scanf (" 1/s" infix).
             push (1#1);
                                                                                                                                                      GRI SAI BHAVANI XEROX
                stracy (infix);
                                                                                                                                                            HIND: 7-1-209/1, Ground Floor,
                 while (infix[i]!='10')
                                                                                                                                                           H.NO. 7-1-209/1, Ground Floor, Theatre, Near Salvam Theatre, Near Salvam
                                         t = infintij:
                                         if (isalpha (t))
                                               pre[j++]=t;
                                                    else
                                                                       if (t== '+' || t== '-' || t== '*' ||
                                                                                                                       t == '/' | t == '/' | | t == '^'
                                                                                                                                                       11 t== '('||t==')')
                                                             Switch(+)
                                                                     (ase ()): push (t); break;
                                                                   Case (+1.
                                                                   Case 1-1:
                                                                                                                                              SRI SAI BHAVANI XEROX
                                                                    case ( %.1:
                                                                                                                                                   H.No. 7-1-209/1, Ground Floor,
                                                                                                                                                Ram Mirra Towers, Near Satyam Theatre,
                                                                   Case 1 /1.
                                                                                                                                                     Ameerpet, Hyderabad-500 016
                                                                 case 1 x1:
                                                                case ' N':
                                                                                                                         ·check ();
                                                                                                                            Push (x);
                                                                                                                            break.
                                                                                                                            while (x!= '(');
```

```
e= i+1:
     uchile (S[xof] != (#')
                                     SRI SAI BHAVANI XEROX
                                      H.No. 7-1-209/1, Ground Floor,
                                     Ram Mirra Towers, Near Satyam Theatre,
                                      Ameerpet, Hyderabad-500 016
        Striver (pre);
        Strrev (infin);
       Printf ("The presin Notation of given enpression
                 evaluation
                                8-f
program to
    #include (stdio.h)
    # included conio. h>
    # include (ctype · h)
    # included math. h>
    # define MAX 50
    char S[MAX Jich;
    float St[MAX], OPI, OPA, OP, val, temp, res;
     ent (=0, j=0, top=-1;
     float pop()
                                       SRI SAI BHAVANI XEROX
        2 return (s[top-
                                        H.No. 7-1-209/1, Ground Float
                                       Ram Mirra Towers, Near Setyam Theatre,
                                         Ameerpet, Hyderabad-500 016
      Void puch (float Val)
        { + + 9 ox }
        S[top] = Val;
```

```
float operate (float op, float opa, char cn)
     switch (ch)
       {
         Case (+): temp = 0P1 + 0P2;
                     break;
         Case (-1: temp = 0P1-0P2.
                      break.
         Case ' * ' : temp = OP1 * OP1;
                      bruak:
         case '/': demp - 0P1/0P2.
                      break.
         case ' 1': temp = pow (op, (int)op);
                      break.
        return temp;
 maines
    christry:
    printf (" Ender the infin
     ; (e, "2.") } mas
      stropy (st, strrev(s));
      while ( stf : ] ! = '\0');
          ch = St i];
           of (isalpha (ch))
              printf ("Enter the value for % c= 4, ch).
              scanf("/f", & val);
              push (val);
          if (ch = = 1+1 | ch = = 1-1 | ch = = 1 / 1 ch = = " / 1"
                           11 ch == 1/1 | ch == 1/1)
```

```
op2 = pop();
   6 P1 = pop();
   res = operate (op, of, ch);
     push (res);
                                   SRI SAI BHAVANI XEROX
                                    H.No. 7-1-209/1, Ground Floor,
                                   Ram Mirra Towers, Near Satyam Theatre,
                                    Ameerpet, Hyderabad-500 016
 8=1+1;
  temp = pop();
  Printf (" The simplified answer for 1.5 =", St);
  printf (" 1/6 of ", temp);
geten();
```

Algorithm-6 post to infix conversion; -

Scan from L	_ to R Repeat 1-2	ROX
Token	Stack Stack	Thealte,
operand	stack St	
operator	pop stack into S1 pop Stack into S2	
	Concatenate S2 operator S1	
	push result on stack	

pop stack to obtain result

AB\$C*D-EF/GH+/+

Token	- Stack	
A	A	
В	A,B	
\$	1.40	161,001,10
C	A\$B,C	Wall Files of
*	A4B*C	Chillian care
٠D	ASBAC, D	SPINA POR TOP STATE OF THE PROPERTY OF THE PRO
_	A \$ B & C - D	Str. M. Wille Log T.
E	A\$B#C-D, B	Ka King
E	4-5 BA C-D, E, F	
/	A\$ B & C - D, R/F	
<i>رب</i>	A\$B*C-D,E/F,G	
+		
+	A \$ B * C - D, E/F, G, H	
/	A & BA-C - D, E/F/9+H	
+	A & B = (= D + G/C// +1)	

Token	stack	
Ą	A	
В	A,B	
C	A, B, C	SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor,
*	A, B*C	Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016
D	A, B*C, D	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
. 6	A, B*C, D, E	
F	A, B* C, D, E	
\$	A, B & C, D, E	
/	A, B*C, D\$	1/E\$F
G	A, B*C, D/	£\$f,G
*	A, B* C, D/	Eff*G
	A, B* C - D	/E\$f *G
H	A, BAC -D	/EAF*9,H .:
*	1)/EIF*G*H
+	A+B*C	D/E4+ * W * H
	A + B * CREATER A BANGER A BAN	OO OLE OLE OLE OLE OLE OLE OLE OLE OLE O
	SEA TOWN TOWN THE	

Algorithm-7

pre to post algorithm :

Scan from R to L repeat 1-2

Token.	operation	H.No. 7 Towers Near Salvam Theates, H.No. 7 Towers, Near Salvam Theates, H.No. 7 Towers, Near Salvam Theates, H.No. 7 Towers, Near Salvam Theates, Near Salv
operand	push to stack	M. No. 7 Tower Near Sod 500
openator	pop stack into SI pop stack into S2	Rammannati
	Concatenate SI Sa Push result on st	

pop. stack to obtain result.

+-* \$ABED//BF+GH

Token	Stack
H	H
$\varsigma_{\!$	H, G
+	OH+ .
F	GH+, F, E GH+, E&F/ SRISAI BHAVANI XEROX GOUND Theatre SRISAI BHAVANI XEROX Ground Floor, Ground Theatre H.No. T-1-2091, Sayam Towers, Near
<u> </u>	4 H+, f, &
	GH+, E&F/ SAI BHA GO Near Salvam 010
/	GH+, E&F/ SRISAIBHAVANIXERO Ground Floor, Ground Floor, Ground Floor, Ground Floor, Fig. 12091,
D	EF/GH+/, D
C	Ef/nH+/, D, C
В	EF/GH+1, D, C, B
Ą	EF/GH+/, D,C,B,A
\$	EF/GH+1, D, C, AB\$
*	EF/GH+1, D, AB&C*
**************************************	EF/GH+/, ABJC*D-
+	ABQC*D-EF/GH+/+

Token	Stack	
•	 	en e
4	H, Cq	60 PM 5 db A 6 ym
F	H, G, F	SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor,
E	H, G, F, E	Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016
\$	H, C1, E&F\$	
D	H, C, EFF, D	
/	HIGID BESF/	
-}t	H, DESCONDERS	Det \$ 10 *
C	H, DEF\$ 10 * 1 C	
В	H, DEF\$/0*, C, B	
*	H, DEf \$ / O +, BC *	
	H, BC * DEF\$/O* -	
⊁ .	BC*DEF\$10*-H*	
A	BC*DEF\$/G*-HA)	
. +	ABC* DEF\$/G*-H*	+

Algorithm - 8
Pre to Infin conversion:

Scanfrom R to L Repeat step 1-2

Token	operation.
operand	Push to stack
Operator	Pop stack into SI Pop stack into S2 Concatenate SI operator 22 Push result on stack

Charles to the state of the sta

Token	Stack
11	H
G	H, C1
+	G+H G+H, F G+H, F, E G+H, E/F GRISAIBHAVANI XEROX G+H, E/F GRISAIBHAVANI XEROX GRISAIBHAVANI XEROX GRISAIBHAVANI XEROX GRISAIBHAVANI XEROX GH-H, E/F GRISAIBHAVANI XEROX GH-H, G-H, G-H, G-H, G-H, G-H, G-H, G-H,
F	CO + H, F, E CO + H, F, E CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, E / F SRI SAI BHAVANI XERO CO + H, NO. 7-1-20911, Ground Floor H.NO. 7-1-2091, Noar Salvani AND HOLDER TO SAI BHAVANI XERO CO + H, NO. 7-1-20911, Ground Floor H.NO. 7-1-2091, Noar Salvani AND H.NO. 7-1-2091, Noar Salvani CO + H, NO. 7-1-2091, Noar Salvani
6	Coth, F, E GOUNT THE Salvam The S
/	9 + H, E/F SRINO. T-1-1-Wers, New York Town Town Town Town Town Town Town Town
1	CHH, F CHH, F, E CHH, E/F SRISAIBHAVANI X Floor, CHH, E/F SRISAIBHAVANI X Floor, CHO, T-1-20911, Ground Theatre, H.No. T-1-20911, Salvam Theatre, H.No. T-1-20911, Salvam Theatre, H.No. T-1-20911, Wall Salvam Theatre, H.No. T-1-20911, Wall Salvam Theatre, H.No. T-1-20911, Wall Salvam Theatre, Ram Mirra Towers, Wyderabad-500 016
Þ	E/F/G+H,D
C	6/F/C1+H,D,C
В	E/F/0+H, D,C,B
A	6/F/G+H, D, C, B, A
4	E/F/O+H, D, C, A\$B
*	E/F/G+H, D, A\$B*C
,	6/F/G+H, A&B*C-D
+	A\$B*C-D+E/F/G+H

	1	4		
Conversion	scon order	Token	operation	
In - bock	L-R	operand	Add to Enpression priority	
		operator	Priority	
In - pre	R-L	1	Add to stack / Delete	
)	Add to Stack / Delete	
post-pre	L-R	operand	push to stack	
post - In	L-R	operator	,	
	GRISA	I BHAVANI XE	lobr, stack ento sa	
pre-pest	R-L- H.No.	I BHAVAIVA 1-1-209/1, Ground F 1 Towers, Near Satyam net, Hyderabad-50/	thealth of a struct of the str	
pre-In	R-L Amee	pet, nydora	Enpression	
			Puch impression on	

Dynamic Memory allocations: >>

- -> As we know that an array is an static memory allocation at the time of compilation we need to define the size once the size has been defined it is never passible either to increase or decrease the size.
- -> To overcome this problem we use dynamic memory allocation.
- -> The dynamic memory allocation will takes place by dynamic functions like, malloc (), calloc(), realloc() and free ().
- -> These dynamic menory functions allocate the memory from the heap area.
- -> when we are allocating the memory by using this functions emplicit we need to perform type castis since return type is void * SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor,

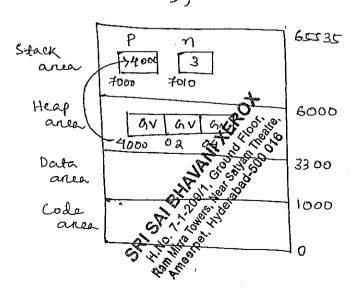
malloc():>

Ram Mirra Towers, Near Satyam Theatre, Used to allocate required number of bytes in memory run time.

→9t takes one argument i.e size in bytes to be allocated. Syntan: -Ptr-Variable = (type_cast *) malloc (sizeof(size)).

en: int * p. int n;

P=(cht *) malloc (n * size of (int));



SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

what is a linked list?

A linked list is a data structure which is collection of o or more nodes, where each node has some information.

- The node will be heterogeneous type which can stone any type of data. Basically each node is divided into two parts, the 1st part contains the information of the element and the and part contains address of the next node which is called as Link.
- The node will be created with the structures concept.

Node

Address of nent node

How to create a node?

Syntan:

Struct node

type 1 info;

type 2 #link;

SRI SAI BHAVANI XEROX
SRI SAI BHAVANI XEROX
H.NO. 7-1-209/11, Ground Floor,
H.NO. 7-1-209/11,

Note: Here struct is a keyword and node is tagname.
The node consider of two fields info and link.

Info: since it contains the information, it can be of type int, float, char, double.

Lg: int info;

char info;

```
Link:
    It contains address of the next node, so, link field
 must be pointer to a node which can be declared as
 Shown below.
            Stouct node + link.
-> The link field must be same type. since we are
 holding the address of a node of the same type. The type
 of structures is called as self refferential
                                             structures.
> By this structures we can hold the address
                                                      int a:
                                                     int *p;
  of a node of same type.
                                                      P = 8a.
 The Structure defination of a node along with
                                                     float s.
                                                      P= & S; X
 declaration can be done as tollows.
          struct node
                                             main()
          int into;
                                             typedet int number.
                                            typedef char of ching,
             Struct node * link;
                                             number a=45;
                                             Storing . C , 1 522 "abe"
           type det struct node * NODE.
  A pointer variable first can be declared as shown
 below
 Method 1; Struct node * first;
 Method 2: NODE
                   first ;
            and link field can be
 The info
                                      accessed using the
 tollowing notations,
 Notation 1: Using
                   * Operator
                                       for Method L
                (* first) · info
                ( * first) · link
Notation 2:
            Using -> operator
                                      method a.
              first -> info
                                    SRI SAI BHAVANI XEROX
               tiret of link
```

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

(:

(

(

(

Applications of Linked list:

-) Linked list concepts are useful to model many different entract data types such as ques, stack and trees.

Advantage of Linked list :>

- -) A linked list is a dynamic data structure and therefore the size of the linked list can be increased or decreased during the enecution of program.
- -y A linked list dokunst require entra space ; so i't doesnot waste entra memory.
- > It provides flenibility in rearranging the items very efficiently.
- The westing <u>limitation</u> of linked list is that it consumes more memory space compared to array, since each node must also contain the address of the next item.
- . > 9t is time consuming and burden process for checking the items in the list.

Types of Linked List :->

Depends on the applications we use different types of linked list like

- 1. Linear Singly Linked List
 2. Circular Singly Linked List
 3. Circular Doubly Linked List
 4. Doubly Linked List.
- -) The general properties on a linked list successive elements are connected by pointers.
- -> Basically the last element points to mull.

The general operations of linked lifet are

1. creation

2. Insertion

3. Deletion

4. Traversing

So Slarching

6. Concatenation

7. Modification etc.

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Arneerpet, Hyderabad-500 016

Linear Singly Linked List :>

By using linear singly linked list we can implement linear data stoucture operations.

program on single linked list

Struct emp

2 int eno;
char ename[20];
Struct emp *link;
3 * start;

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Amearpet, Hyderabad-500 016

7/12/2016

790 a single linked list a node has minimum a fielde information field and the link field.

- -> Every node will link to the other node by placing the address of the nent node in the previous node of the link field.
- The last node of the link field indecating by NULL, that indecates end of the node.

Both	· Link	Info	Link		
10	2000	20	7000	30	~
1 000		2000	•]	3000	,

```
#include (stdio.h)
 Struct emp
  3
                               SRI SAI BHAVANI XEROX
      . Ent eno;
                                H.No. 7-1-209/1, Ground Floor,
                                Ram Mirra Towers, Near Salyam Theatre,
      Char ename[20].
                                  Amesipet, Hyderabad-590 016
      Struct emp * link;
   3 * start ;
Creation ()
  Struct emp * tmp, *9;
  tmp = (struct emp #) malloc (size of (struct emp);
   print (" Enter the eno; ");
                                           tmp = 1000
   scanf (" xd , &tmp -> eno).
                                           Start = M
                                                1000
                                                           ename link
  Print (" Enter the Ename: "):
                                                           Scott
  scanf (" 1/2", &tmp -> ename);
                                                     1000
  tmp -> Link = NULL;
   if (start = = NULL)
   · Start = tmp;
                                                 eno enone link
   else
                                                      Seatt N
      9 = start :
      while (q -> link |= NU
                                             tmp=2000
                                                         4171
                                                          2000
                                               for third
                                                         rode
                                             enane link
                                          191 Scott 200
                                          1000
                                    tuf=3000
                                               171 Clark M300
                                                                Stives N
                                    Start=1000
                                                             기 | III
 7 // end of creation.
                                               2000
                                                             3000
                                     9 2 1969%
```

```
Void display()
     Struct emp + 9:
     q=start;
    while (9%-NULL)
                                            SRI SAI BHAVANI XEROX
                                             H.No. 7-1-209/1, Ground Floor,
                                            Ram Mirra Towers, Near Satyam Theatre,
      print (/1 /d /.s
                                             Ameerpet, Hyderabad-500 016
    if (q = = NULL)
        print ("In list is empty");
    عداه
     ş
        printf ("In The data in the list "In").
     while (91=NULL)
       printf (" | 1/d 1/s (1/d) | 1/d | -->", 2->eno, 2->ename,
        9=9-> link:
                               GRI SAI BHAVANI XEROX
                                H.No. 7-1-209/1, Ground Floor,
                                Ram Mirra Towers, Near Satyam Theatre,
                                 Ameerpet, Hyderabad-500 016
addat beg()
  Struct emp * emp.
    tmp = (struct emp *) malloc (size of (struct emp));
    print (" Enter the eno: ");
    scanf ("%d", & tmp -> eno),
    Printf ("Enter the Ename: ").
    scanf (" " d", & trup -> ename);
                                                         2000
      tmp -> link = Start;
      Start = tmp;
```

```
addafter (int pos) 11 pos=6
                                                                                                                                                                                                           DO0 = 6
        Struct emp * 9, * tmp.
         int is
                                                                                                                                                                                9=4,000
                                                                                                                                                                                                                            · 12 p <5
                                                                                                                                                                                                                                           X <5
          92 Start
                                                                                                                                                                                         1,000
                                                                                                                                                                                                                                          215
                                                                                                                                                                                      2,060
                      for (120; 14 pos -1; 1++)
                                                                                                                                                                                          3000
                                                                                                                                                                                                                                           3<5
                                                                                                                                                                                           N COOP
                              1
                                        9 = 9 -> link;
                                           4 ( 9 = = NULL)
                                                       printt ("There are less three OX of elements", pos).
Tetum,

Ground Floor,

Ground Floor,

Ground Floor,

Resident (" Pos).
                                                                                                                                  7.1.209/1. Ground Floor.
                                                                                                                   HING. 1-7-20917, Ground Flour, of HING. 1-7-20917, Ground Flour, of HINARIAN NAS. HINARIANAL-ENN NAS. AMBORNO, HINARIANAL-ENN NAS. AMBORNO, MARINARIANAL-ENN NAS. AMBORNO, MARINARIANA 
                                                                                                                        ati Wills Iomete' Meet 29thau Tueste
While Iomete' Meet 29thau Tueste
While Iomete' Meet 29thau Tueste
                                      311 for
          top = (struct emp *) malloc (size of (struct emp));
              Printf (" Enter the eno: "),
                                                                                                                                                                    for
                                                                                                                                                                                   26 209
             seant ( u % d ", & tmp -> eno);
             Printf (" Enter the Ename: ");
                                                                                                                                                   171 Leing 1000
            scanf (" 1.5", & tmp > ename).
                                                                                                                                                                                               > 1.71 Scott 2000
                                                                                                                                                     4000
              tmp->link =q-> link;
                    g > link = tmp;
                                                                                                                                                                                                                                         121 YOUN
                                                                                                                                                                                            100 NK1 300
                                                                                                                                                        Pos: 2-1
                                                                                                                                                             /O'< 1
                                                                                                                                                                                            ५०००
                                                                                                                                                               1<1
       del (int data)
                                                                                                                                                                                    921000
                                                                                                                                                                                             tmp = 5000
              Struct emp *tmp, *q;
                 if (start -> eno == data)
                                                                                                                                            1000
                                                                                                                                                                                $ 000
                                                                                                                                                                                                                          2000
                         tmp = start ;
                            Start = Start -> link; // first element
                             tree (tmp);
                                                                                                                                                              Stmp = 4000
                             return;
                                                                                                                                                                 Start = 1000
                                                                                                                                                                         tree (4000)
```

 C_{i}

```
9 = Start;
  while (9-> link; = NULL)
     {
if (9-> link->eno = = data) // element deleted in
beti
                                                            between
          tmp = 9 -> link:
                                               Void deletelier
           9-) link = tmp -> link;
                                                  drest emp & days
           tree (tmp):
                                                  while for a first
                                                       Amps Stores
                                                       The was believe.
                                                      2 + > ( ( ( mp) ) 4
                                                      Fit ( Success justing
                                                                die telen
     print ("Element ", I not found In", data),
                                                                  3.77
main()
Ent choice, i, n, pos, ele
                                         SRI SAI BHAVANI XEROX
                                          H.No. 7-1-209/1, Ground Floor,
  start = NULL;
                                         Ram Mirra Towers, Near Satyam Theatre,
                                          Ameerpet, Hyderabad-500 016
  Clascac):
   while (1)
    printf ("In 1. create a list In");
                                                                printf (" |n 2. Display" |n");
    printf (" In 3. add at beg In");
    printf ("In 4. add after In");
    print ("In a. deletation In").
    printf ("In b. Enit "):
     printf ("In Enter the choice:");
     scanf (" y.d", & choice),
     Switch (choice)
        Case 1: printf (" In How many nodes do you want ");
                  s canf ("Y.d", &n).
                   for(i=0;i(n;i++)
                   Creation ();
                                             SRI SAI BHAVANI XEROX
                    break;
                                              H.No. 7-1-209/1, Ground Floor,
                                             Ram Mirra Towers, Near Salyam Theatre,
                                              Ameerpet, Hyderabad-500 016
```

SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, case 2: display(); Ram Mirra Towers, Near Salyam Theatre, Ameerpet, Hyderabad-500 016 break; case 3: addatibeg(); breaks printf (" Enter the position 3"); Scarif (" Y.d", & POS); addafter (pos); break; Case 5 : printf (" Enter the empno to delete: "); Scanf (" ", d", & ele); del (ele); break case 6 : Enit (0). } llend of switch. } // end of while. getch(); SA! EMAYAMI AERUA Floor, Ground Theatre, H.No. 7-1-209/1, Ground Theatre, H.No. 7-1-209/1, Warshad-500 018 Ram Mirra Towers, Nearabad-500 018 Ram Marner, Hyderabad-500 018 ? I end of main. am mina lowers, Near Salyam Ineana Ameerpet, Hyderabad-500 018

```
#include (stdio.h)
Struct node
    ent enfo;
    Struct node * next;
                                      SRI SAI BHAVANI XEROX
                                        H.No. 7-1-209/1, Ground Floor,
                                       Ram Mirra Towers, Near Satyam Theatre,
 ] * start;
                                        Ameerpet, Hyderabad-500 016
 Void create ()
    struct node * new, *t;
  new = (struct node *) malloc(size of (struct node)).
  print (" Enter the information: "):
   scanf (" /d", & new -> info);
   new-Inent = NULL;
      if ( start == NULL)
         Start = new;
                                 10 2000 720 3000
                                                 30/4000/
                                          2000
                                                  3000
                                                                 GOOL
          t= start
           while ( + -> nent ! = NULL)
                                                 SRI SAI BHAVANI XEROX
                                                  H.No. 7-1-209/1, Ground Floor,
   Void display ()
                                                 Ram Mirra Towers, Near Satyam Theatre,
                                                  Ameerpet, Hyderabad-500 016
      9 = start;
       of (q == NULL)
            printf ("The lixt is empty");
```

```
while (91= NULL)
                                               SRI SAI BHAVANI XEROX
           printf ("Yd", 9 > into).
                                                 H.No. 7-1-209/1, Ground Floor,
                                                Ram Mirra Towers, Near Satyam Theatre,
                                                 Ameerpet, Hyderabad-500 018
          29=9->nent;
  Struct node * reverse (struct node * p) // P= 1000
       Struct node #9, * 8:
         9 = NULL.
                                          1000
                                                 2000
                                                           3000
                                                                    4000
         while ( P) //1000
                                102000
                                       20 3000 30 4000
                                                       7/40/59/00
                                                                MOJK
                                1000
                                                        4000
                                                                 5000
                                          P= 10,60 3,000 4,000 roup
            P->nent = 2:
                                          9 = N 1960 2000 3000 4000 5000
             9 = P.
                                          8 2 2,000 3,000 40,00 5700 N
            return q.
  Struct node * secrev (struct node * P)
                                                  1000
                                                          2000
                                                                  3000
2
          struct node * head
                                                                          4000
3
                                       10 2,000 720 3060
           of (1P)
                                                                          M
4
                                               2000
                                        1000
                                                               4000
                                                                       000
ζ.
           return NULL:
          if (P-) ment)
                                                        neevere
                                                      head
                                                            5000
Q.
             head = recrev (p-) nent).
                                                       neeneves
                                                       read
                                                            4000
                                                       5000
              P-> nent -> nent = P.
              P-> nent = NULL.
                                                       head
                                                            3000
              return head "
                                                       5000
.
                                                       recrues
                                                       head
            else
                                                       5000 2000
              return p.
                                                      TRETENC 1
                                                      head
                                                      5000 1000
                                                 18
                                                       maine)
                                                      Stant
                                                       ~5000
```

```
main ()
  I get choice, i', n,
                 Start = NULL;
                   C) 1221/D
                   while (1)
                             print ( "In 1. create a list in ");
                               print ("In 2, Display In");
                                Printf ("In 4. reverse HAVANI XEROX

Printf ("In
                               Printf ("In S. Extender Hyderabad-500 016)
Switch (Choire)
                                       Case 1; printf ("In How many nodes do youwant's)
                                                                                          s cont (" /.d", 20);
                                                                                            for(i=0; i(n; i++)
                                                                                               creation ();
                                                                                              break ;
                                        Case 2: display ();
                                                                                              break
                                                                                              Start = reverse (Start);
                                                                                              brusk;
                                      Case 4: Struct necese (start);
                                                                                               break
                                      Case S: Enit (0);
```

```
for single linked list and display
a) write
             a program
                    from last?
   the 1th node
  Struct
           Node
                                        SRI SAI BHAVANI XEROX
                                          H.No. 7-1-209/1, Ground Floor,
    Ent date ;
                                         Ram Mirra Towers, Near Satyam Theatre,
                                          Ameerpet, Hyderabad-500 016
       Struct Node * rent 9
   I inserting node at starting for simplicity
    CreateNode (struct Node ** head, int data)
         Struct Node # temp = ( struct Node *) malloc (size of
                                                            ( struct mode)).
          temp -> data = data.
          temp -> rent = * head .
           * head = stemp.
                                              200 300
                                                                 140,600
                    H.No. 7-1-209/1, Ground Floor,
                                                200
                                                                   400
                   M.NO. 1-1-2031, Gradin Flouring Theatre, Near Satyam Theatre, Ram Mirra Towers, Near Satyam or se
                     am Mirra Towers, Near paya''' 016
Ameerpet, Hyderabad-500 016
                                                N70K
                                                       HOD) $0
                                                                   \sigma\sigma
                                                 Void display (struct
                               Mode * head)
           while ( head )
              printf ("y.d", head -> data).
              head = head -y next;
  Il Finding the 19th in singly linked list from finst
     Void find nuth Mode (struct Mode + head, int n) 1100 3
       int count = 1;
          Struct Node & temp = head ;
           if ( head = = NULL)
```

```
printf (" Lest is empty \n");
    while (count < n 94 temp)
       temp = temp -> next;
                                       SRI SAI BHAVANI XEROX
        count ++;
                                        H.No. 7-1-209/1, Ground Floor,
                                       Ram Mirra Towers, Near Satyam Theatre,
                                        Ameerpet, Hyderabad-500 016
      if (count == n)
                  The required Node at the location
                          is y.d \n", o, temp -> olata).
        return o
                                Nodes
                                         are present in the
       print ("Less NO
                                           linked like ):
Il finding the Neh Node in single linked list from last
      Find Neh Node from Last (struct Node of head, int n)
                                                               1100 3
      count "
                                                              100 5
  struct Node * temp = head , * Nth node = NULL.
  if ( head = = NULL)
      prinef ("List is empty");
    for ( count = 1; count <n;
        if (semp) Him 200
                                      SRI SAI BHAVANI XEROX
                                  11 300
                                       H.No. 7-1-209/1, Ground Floor,
                                      Ram Mirra Towers, Near Satyam Theatre,
                                        Ameerpet, Hyderabad-500 016
```

```
while (temp) / 100 (000
  {
if (Nthrode = = NULL)
                                    SRI SAI BHAVA'NI XEROX
                                     H.No. 7-1-209/1, Ground Floor,
      Nthrode = head , 11,00
                                     Ram Mirra Towers, Near Satyam Theatre,
                                     Ameerpet, Hyderabad-500,016
      else
      Nthrode = Nthrode -> next;
      temp = temp -> nent;
   if ( Nthrode )
     printf ("The required Node at the location yed is
                     y.d m", n, Nthnode -> data).
   elle
     print ("Less No: of Nodes are present in linked
                                                     C" teil
main()
          Mode & head = NULL;
   int or ;
   createrode ( & head , 70) .
                                                   head = 100
   create Node ( & head, 60);
                                                    n = 5
   Create Mode ( Chead, 50).
   createrrode (& head, 40).
   cruateriode (8 head, 30)
    ereaterrode (9 head, 20)
    Create Node (qhead, 10).
   Printf ("Elements
                      in the linked list
    display (head).
    print ("\n");
    printf ("Enter the position of required node: \n");
    scanf (" y.d ", &n);
    find NAhnode (head, n).
     Print ("In");
```

(·

(

printf ("Enter the position of required node to find from last: mi). scand (" 1.d", &n); find NH Node from Last (head, n): return 0; SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016 Write a program swapping the kth node data from to the beginning and leth node data from last in the single linked list. Void Swap (struct Node + head, int k) Struct Node * tmp1 = head, * tmp2 = head, *tmp3= if (head = = NULL) { printf (" List is empty"); for (count = 1; count (K; count ++) } if (tmp 1) tmp1 = tmp1 -> nent; SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, tmp2 = tmp1 Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016 while (tmp 2)

```
P = tmp 1 -> data.
          tmp1 -> data = tmp3 -> data.
                                  SRI SAI BHAVANI XEROX
          tmp3 -> data = P:
                                    H.No. 7-1-209/1, Ground Floor,
                                   Ram Mirra Towers, Near Satyam Theatre,
                                    Ameerpet, Hyderabad-500 016
13/12/2016
 Void swap (struct rode * *head, int K)
    Struct node *ptr, *Kthnodefom beg, *Kthnodefomlet,
     int count = 1, temp;
     Kthnodefron let = Kthnode from beg = * head;
      if (K = = 0)
       printf ("There is no modes
                                      to be display \n").
                                         SRI SAI BHAVANI XEROX
       enit (0).
                                           H.No. 7-1-209/1, Ground Floor,
                                          Ram Mirra Towers, Near Satyam Theatre,
                                           Ameerpet, Hyderabad-500 016
Il finding the kth node from begining
  while (kehnode from beg & q count < K)
     count ++;
     Kthnodefrom beg = kthnode from beg ->next ;
 Il if the while loop terminates due to NULL then
    . Ef (! kthnode tombeg)
       printf ("There are less No: of Modes in the list 10")
      enit(0);
   PEr = Kthnodetombeg => nent ;
                2 pointer at a time
                                        . If ptr reaches last
  / NOW move
                   kshnode from let
                                        points to the kth
    position
     node from the last
```

(

```
while (ptr)
    pto = pto-ynext;
    Ket nodefrom let = Kethnode from let -> next ;
                                 and kethnodeform. let using
I swapping Kehnodefrom heg
temporary variables temp
                        the data only.
I here we swapping
  temp = kehnode from beg -> data;
  Kthriodefombeg -> data = Kthrodefomlet -> data;
  Kthnodefom lst -> data = temp.
  main()
                                             SRI SAI BHAVANI XEROX
   Struct Node + head = MULL;
                                              H.No. 7-1-209/1, Ground Floor,
                                             Ram Mirra Towers, Near Satyam Theatre,
    int Kg
                                              Ameerpet, Hyderabad-500 016
    Chrscr().
    createrrode (& head, 70).
     create Mode (& head, 60)
     createrode (q head, 50).
     create Mode (8 head, 40),
     criateriode (ghead, 30)
     createriode (& head, 20).
     createrisde (ghead, 10).
      printf (" Enter the K value: ").
      scanf (" /.d", & K);
      printf ("In Before swapping In");
      display (head);
      printf ("In After swapping In")
       swap ( & head, K);
                                 SRI SAI BHAVANI XEROX
       display (head);
                                   H.No. 7-1-209/1, Ground Floor,
                                  Ram Mirra Towers, Near Satyam Theatre,
                                    Ameerpet, Hyderabad-500 016
```

```
Enter the k Value; 3
   Before swapping
         20 30 40 50 60
   After swapping
         20 50 40
                      30 40
Q) Given a null terminated
                                           15. Hellod hode, 5th rode)
  nodes into two lists (first 1864)
and Eath node, 6th node HIND
    Sample 2/0 formals
   Enter the input linked link values
   Enter the value
   Do you want to add another mode [Y/N]
   Enter the Value
      3
     90
   The elements in the linked list are: 5
   The elements in the 1st linked list are; 5 10
   The elements in the and linked list are: 8
     struct node
      3 Ent data;
                                      SRI SAI BHAVANI XEROX
          Struct node * next ;
                                        H.No. 7-1-209/1, Ground Floor,
                                       Ram Mirra Towers, Near Salyam Theatre,
      } * thead = NULL;
      Struct node * head of list 1 = NULL Ameerpet, Hyderabad-500 016
       Struct node # headoflot 2 = NULL.
      Void arrangelist ()
           struct node * temp = head , * list1 = NULL, * list2 = NULL,
            struct node * ptr =
```

```
Struct node
      Struct node * newt.
   I * head = NULL:
Creation ()
                                                   SRI SAI BHAVANI XEROX
  Z
                                                    H.No. 7-1-209/1, Ground Floor,
                                                   Ram Mirra Towers, Near Satyam Theatre,
     char che
                                                   Ameerpet, Hyderabad-500 016
     do
       struct com node # tmp , # 9;
       tmp = (struct node *) malloc (size of (struct node)).
       printf (" Enter the data : ").
       Scanf (" ", & tmp -> data).
       trop -> next = NULL:
         if ( head == NULL)
            head = tmp.
           2 2 head.
            while (9) rent 1 = NULL)
              9 = 9 > nent.
               9 > nent = tmp;
          Printf (" Do you want to add another node: In");
              ch = getenec);
          I while (ch != 'n');
         while (tmp!=NULL)
                                                 SRI SAI BHAVANI XEROX
              try = try -> rent;
                                                   H.No. 7-1-20911, Ground Floor,
                                                  Ram Mirra Towers, Near Salyam Theatre,
                                                    Alligatbat, Hyderabad-500 016
```

```
Struct the * headoflet 1 = NULL.
Struct node of headoflet 2 NULL:
Void
        Rearrange ()
            node * temp = head, * list1 = NULL, * lista = NULL,
   struct nade #ptr.
    Ent 1=1;
                                     SRI SAI BHAVANI XEROX
     while (temp } = NULL)
                                      H.No. 7-1-209/1, Ground Floor,
       Z
                                     Ram Mirra Towers, Near Satyam Theatre,
                                      Ameerpet, Hyderabad-500 016
          if( & 1/2=21)
              Ptr = (struct node *) malloc (size of (struct node))
              ptr -> data = temp -> data;
               pto -> nent = NULL:
              if (head of let 1 = 2 NULL)
                  headoflet 1 = ptr.
                 list 1 -> nent = ptr;
                                 SRI SAI BHAVANI XEROX
                                   H.No. 7-1-209/1, Ground Floor,
                 lists = ptr;
                                  Ram Mirra Towers, Near Satyam Theatre,
                                    Ameerpet, Hyderabad-500 016
         else
               ptr = (struct node #) malloc (size of (struct node))
                Ptr->data = temp->data.
                 .Ptr > & nent = NULL.
                  if (head of let 2 = = NULL)
                    headoflet 2 = ptr;
                else
                  list 2 > nent = pto;
                    List 2 = ptr.
                temp = temp ->nent
```

```
main()
      Struct node * head = NULL;
     Printf (" In Enter the Enput linked list values 1");
      Creation ():
      Printf ("Before arrange list is:");
       display (head);
       Rearrange ();
       printf (" The elements in the 4st linked list are:").
        display (headoflota);
        printf (" The elements in the and linked list are: ").
                                     SRI SAI BHAVANI XEROX
        display (headoflot+);
                                       H.No. 7-1-209/1, Ground Floor,
                                      Ram Mirra Towers, Near Salvam Theatre,
        return o;
                                        Ameerpet, Hyderabad-500 018
write a program to delete the node
                                                  without using head
 pointer in a single linked list.
  for this problem we need to know the actual address of
the required node, hence we need to save the address
 by using a global variable. In this program suppose is any
saving the 3rd node address from last.
         void delete (stouct node * nodestodel)
               stout node * nentrade;
                 if (nodetodel = = NULL)
                 nentrode = node todel -> nerdo
                 node to del -> nent = nent node -> next
                                      SRISAI BHAVANI XER
                                        H.NO. 7-1-209/1 Noor Environ Thooks
                 & free (nentrode);
                                       H.NO. 1-1-20911, Ground Floor, Andrew H.No. 1-1-20911, Ground Floor, Andrew H.Norrahad. Find B16
                                         an wing towers, wear salvain linging
                  return.
```

```
14/12/2016
                                                                                                                         SRISAI BHAVANI XEROX
            #include ( St dio. h)
                                                                                                                                H.NO. 7-1-20911. Ground Floor.
                                                                                                                                H.NO. 1-1-2091 I. Ground Flour.
H.NO. 1-1-2091 I. Ground Flour.
Ram Miria Towars, Near Salyam Theatre.
Ram Miria Towars, Near Salyam Or B
         # include ( conio.h)
                                                                                                                                      am wulgan Hyderabad-500 01$

Westerpet, Hyderabad-500 on the second seco
        # include (stdlib.h)
         Struct node
                ? Int data;
                          Struct node * rent "
          Struct node # ptr1;
            int count:
        1/ execte a Mode
     void createnade (struct node * * head, ent data)
               struct node # new node = (stouct node #) malloc (size of (stough
                  count ++.
                    if (count = = 3)
                                    per1 = newrode;
                                      newrode -> data = data.
                                       newnode -> nent = *head.
                                      * head = newnode.
          void display (struct node & head)
                   } if (head == NULL)
                                mile (head)
                                            printf (" ", d \ t", head -> nent).
                                           head: head -> nent;
```

```
Il Deleting the mode without using head pointer.
Il we need to copy the next node information into current
 I node and we are actually deleting the next node
 I ( Since we have all the information of next node in
I current node)
Void deletenade (Struct nade # pts)
                                     SRI SAI BHAVANI XEROX
   Il saving the nent node address. No. 7-1-209/1, Ground Floor, Near Satyam Theatr
    etruct. node of temp = ptr -> newtameerpet, Hyderabad-500 016

Ptr -> data = Ne- .
    Ptr -> data = ptr -> nent -> data;
    // printf (" y.d \ #", ptr -> data),
       pto->nent = pto->nent -> nent;
    1/ printf (" /d /t", per -> nent).
      free (temp).
                                     730 400
                                           740,500
                                      300
    ent main ()
       Struct node # head = NULL;
                                               Ptr = 500
       Createnade (I head, 70).
                                                4mp = 600
       create node (8 head, 60).
       Create node (9 head, 50).
       createnode (& head, 40).
       createrode (& head, 30).
        Createnode (8 head, 20)
       createnade ( & head, 10):
        probable ("Display the node before deletion \n ");
        display (head);
    I calling the delete node function
                                          with seved argument
      deleterode (ptr1);
      printf ("In Display the node after deletion "In").
      display (head);
      return o.
```

SRI SAI BHAVANI XEROX

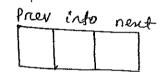
H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

As in the single linked list the traversing of data will takes place only one way inorder to traversing the data we are in two ways i.e forward and backword, we use Double linked list.

non-linear data structures.

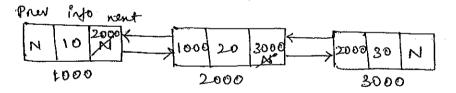
-> In double linked list a node has 3 fields.

(i) Information (ii) previous field (iii) Nent field.



-> In double linked list the nent tilld of the node contains nent node address, the previous field of the node contains before node address. This makes double linked list.

-> The first node of the previous field and nent field of the last node contains NULL.



struct dll

Struct all * prev;

Ent Ento;

Struct all * rent;

I * start;

Creation()

SHING TO SHING TO SHING TO SHIP TO SHI

Struct all # temp, # P

tmp = (struct ell *) malloc (size of (struct all));

printf (" Enter the information");

Scanf ("Y.d", A temp -> info);

1000

, Jemp 22000 Sport = 1000 temp -> prev = NULL. temp -> next = MULL: 20 if (start = = NULL) 1000 2000 start = temp. -clse P = start : while (P-> nent 1 = NULL & && P -> PREN 1 = NULL) . P->nent = temp; SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016 temp -> nent = NULL; if (start = = MULL) temp -> prev = MULL. else P= Start

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor,
Ram Mirra Towers, Near Satyam Theatre,
Ameerpet, Hyderabad-500 016

p.

```
# in cludes stdio.h)
Struct dll
                                SRI SAI BHAVANI XEROX
  Struct all * prev;
                                 H.No. 7-1-209/1, Ground Floor,
                                Ram Mirra Towers, Near Satyam Theatre,
   int info;
                                  Ameerpet, Hyderabad-500 016
  Struct all # nent ,
 } * stant;
Creation()
 Struct all # new, *p,
new=(struct dll #) malloc(size of (struct dll)),
Printf ("Enter the information");
                                                  Prev
                                                       info
Scanf (" y.d", qnew -> into);
                                                             neno
                                                        10
new -> next = NULL.
                                                       1000
if (Start = - NULL)
                                           Start = 1550
                                      for and node
      Start = new;
     new-yprev = NULL;
                                                                 20
                                                              مومل
 else
                                             P=1000
    P= start;
                                              new = 2000
                                                          3000
                                    1000
                                                     2000
                                   P=1000
    new-prev=p.
                                                           ತುಣ
                                                               30
                                                                     N
                                    new = 3000
                                                              3000
```

```
vold display()
           9= spart;
                                                                                                                                                 SRI SAI BHAVANI XEROX
                                                                                                                                                     H.No. 7-1-209/1, Ground Floor,
          if (q == NULL)
                                                                                                                                                Ram Mirra Towers, Near Satyam Theatre,
                                                                                                                                                   Ameerpet, Hyderabad-500 016
           Printf (" List is empty ");
    else
              Printf (" Data in the list!"):
            while (9!=NULL)
                   Printf (" \t 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 - 1/4 -
                            = 9 -> nent;
                                                                                                                                              SRI SAI BHAVANI XEROX
                                                                                                                                                   H.No. 7-1-209/1, Ground Floor,
addat beg ()
                                                                                                                                                Ram Mirra Towers, Near Satyam Theatre,
                                                                                                                                                  Ameerpet, Hyderabad-500 016
        Struct all * tmp.
Imp = (struct del *) malloc (size of (struct del)).
  Printf ("Enter the Information");
  Scand (" / d", & tmp -> info),
   tmp->nent = start;
                                                                                                                          4,000
       Start -> prev = tmp;
                                                                                                                                   [ 0
                                                                                                                                            2000
       Start = tmp.
                                                                                                                                         000
    2 tmp -> prev = NULL;
                                                                                                                                                                               2000
                                                                                                                                                                                                                          3000
                                                                                                          40
                                                                                               Ν
     addafter (int pes)
                                                                                                                                                    tmp = 4000
                                                                                                            4000
                     Struct all *tmp, *9;
                       int it
                                                                                                                SRI SAI BHAVANI XEROX
                      2 = Start;
                                                                                                                     H.No. 7-1-209/1, Ground Floor,
                                                                                                                   Ram Mirra Towers, Near Satyam Theatre,
                                                                                                                        Ameerpet, Hyderabad-500 016
```

for(=0; ix pas-1; i++) SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, $\begin{cases} \end{cases}$ Ram Mirra Towers, Near Satyam Theatre, 2 = 2->nent" Ameerpet, Hyderabad-500 016 84 (9 == NULL) Printf ("There is less no of % of elements:", pos). return , tmp = (Struct dll *) melloc(size of (struct dll)). Printf ("Enter the information"). scanf ("xd", & tmp-yinfo; 4000 4 2000 200030 3000 10 1000 20 trop -> nent = 9 -> nent; 2000 1000 3000 9-> nent = tmp; tmp-yprev=q. y tmp -> nent -> priev = trup. 9210002000 4000 del (int data) 0 < 2 - 1 2 1 Imp = 4000 Struct all #tmp, #9; ** it (stort -> info == data) 3000 tmp = start = 10 2,000 30 Ŋ Start = start -> next: 1000 3000 Start -> prev = NULL: free (tmp). Hores data = 20 return 9=1000 9=start; tmp= 2000 free (2000) into==data) tmp = 9 ->nent 9-Ynext = tmp -> next: trop -> next -> newsporev=q. free (tmp); z nestum g

```
9 = 9 -> next
 printf ("Element not found");
main ()
 int choice, i, o, pos, ele;
                                       SRI SAI BHAVANI XEROX
                                        H.No. 7-1-209/1, Ground Floor,
Stanf = NULL o
                                       Ram Mirra Towers, Near Satyam Theatre.
                                        Ameerpet, Hyderabad-500 016
claserc).
while (1)
  Printf ("In 1 · create a live");
 printf (" |n 2. Display ");
 Printf (" | m 3. Add at beginning ");
 print ("In 4. Addafter"):
 print ("In S. Delate");
 Printf ("In 6. Enit").
 printf ("Enter the choice");
                                           SRI SAI BHAVANI XEROX
 scanf (" /d", &choice);
                                            H.No. 7-1-209/1, Ground Floor.
                                           Ram Mirra Towers, Near Satyam Theatre,
  Switch (choice)
                                            Ameerpet, Hyderabad-500 016
   Ę
  Case 1 i prints ("In How many nodes do you want:");
            scanf (""/d", &n);
            for (i=0; i(n; i++)
             Crccation ().
             break:
 case 2; display ();
 case 8; addatheg().
 case 4: printf ("Enter the position");
            scanf ( " y.d", & pas);
            addagter (pos).
            break,
```

```
Care 5: print ("Enter the data to delete").
                                                 s canf (" y.d", & ele);
                                                  del (ele).
                                                 bruak :
                                             enit (o).
                                                                                                                          SRI SAI BHAVANI XEROX
                                                                                                                               H.No. 7-1-209/1, Ground Floor,
     getch();
                                                                                                                              Ram Mirra Towers, Near Satyam Theatre,
                                                                                                                                   Ameerpet, Hyderabad-500 016
      Struct node
               2 Int data;
                        Struct node & prev.
                       Struct node * nent;
             } * head, *lack;
                    head 2 NULL ;
                    last = NVLL.
 deletefombeg ()
                        struct node * todelete:
                               todelete = head.
                                  head = head -> nent
                                   head -> prev = NULL;
                                          tner (todelete);
                                                                                                                                                                             SRISAI BHAVANI XEROX
                                                                                                                                                                                    H.NO. 7-TANDER NACE CANDAM THOSE
                                                                                                                                                                                    H.NO. 1-7-20917, Ground Floor, Arno Nina Towers, Near Salvan Theale, H.No. 1-7-20917, Ground Floor, Arno Nina Towers, Near Salvan Arno Nina Towers, Near Sal
      delete from end ()
                                                                                                                                                                                          am wire lowers, near sayam incare
Americal Applerabad, 500 01 &
                 2 struct node * todelete;
                                         todelate = last;
                                           last = last , -> Prev;
                                             last -) rent = NUL
                                              free (to delete).
```

```
deleterfrom N (int pas)
E struct node * top ;
    tmp = head;
    for ( 121; i/pas 28 tmp ! = NUZL; i++)
    if (pas ==1)
                                 SRI SAI BHAVANI XEROX
                                   H.No. 7-1-209/1, Ground Floor,
                                  Ram Mirra Towers, Near Satyam Theatre,
                                   Ameerpet, Hyderabad-500 016
     if (top == las
         trop -> nent -> prev = trop -> prev,
          free (tmp);
        } Pf ("in valid position").
```

H.NO. 7 Toware Near Canvan Theat H.NO. 7-1-209/1, Ground Floories am mira Towers, Near Selvan Theatre

```
SRI SAI BHAVANI XEROX
del (int num)
                                    H.No. 7-1-209/1, Ground Floor,
                                   Ram Mirra Towers, Near Satyam Theatre.
2 struct node * tmp, * 9.
                                    Ameerpet, Hyderabad-500 016
    if (start -> info == num)
        tmp = start .
                                / # first element deleted */
        Start = Start -> nent;
        Start -> prev = NULL;
        free (tmp);
         neturn;
     9 2 Sterrt.
     while (q-) rent -> nent ] = NULL)
                9->nent=tmp->nent;
                                                En between
                 trop->nent->prev=9:
                 free (tmp);
              929->nent;
              9->nent = NULL
              neturn
```

Circular Singly linked list:

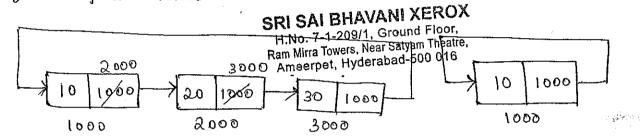
The drawback of doubly linked list is, there is a wastage of memory location of two fields, first node of the previous field and next field of last node.

→ To øvercome this problem, we use Circular linked little.

→ In circular singly linked list, a node has minimum two
tields. (i) information

(ii) Nent tield.

Info ment 3 of is almost similar to single linked list; The next Field of the last node contain the first node address.



Struct node

int info;

Struct node # link;

I * last;

Stouct node *new, *P;

enewtion ()

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

new= (struct node *) malloc (struct (stouct node)),

Printf (" Enter the information"),

Scanf (" / d", & new -> info),

Thew-> dink = new;

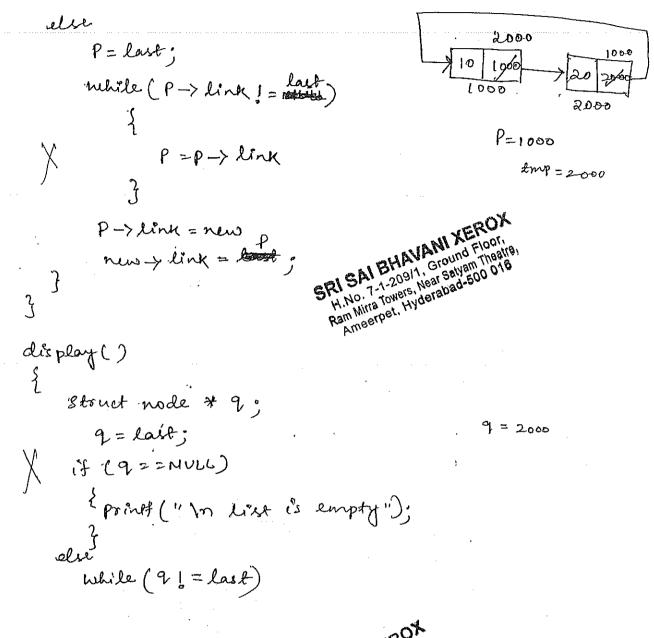
if (last == NULL)

Last = new;

last = new;

new-> link = new;

SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Salyam Theatre, Ameerpet, Hyderabad-500 016



SRI SAI BHAVANI XEROX
SRI SAI BHAVANI XEROX

Ground Floor

Thear Salvan Thear

Towers, Near Salvan 500 016

Ram Mirra Towers, Hyderabad 500 016

Ram Mirra Towers, Hyderabad 500 016

```
Struct node
       Estruct node * link;
                    ent num;
         3 *last;
                                                                                                                                         SRI SAI BHAVANI XEROX
Creation (int num)
                                                                                                                                               H.No. 7-1-209/1, Ground Floor,
                                                                                                                                          Ram Mirra Towers, Near Salyam Theatre,
                                                                                                                                              Ameerpet, Hyderabad-500 016
                  Struct node * tmp, * 9;
                tmp = (struct node *) malloc (size of (struct node));
                   tmp => info = num.
                        if (last == NULL)
                                      { last = tmp;
                                                                                                                                                                                                  2000
                                             tmp-> link = tmp;
                                                                                                                                                                                               todo
                                                                                                                                                                                10
                   else
                                                                                                                                                                                   1000
                                                                                                                                                                                                                                                 2000
                      } tmp -> link = last -> link;
                                                                                                                     SRI SAI BHAVANI XEROX
                                 last -> link = tmp;
                                                                                                                                                                                                                                       last = 1,000
                                                                                                                         H.NO. 17-1-20911, Ground Floor, H.NO. 17-1-20911, Ground Floor, Oracle Ram Wirts Towers, Near Salvam Oracle Ram Wirts Towers Hurtershad-Food Oracle Ram Wirts From Hurtershad-Food Oracle Ramanana Hurtershad-Food Oracle Ramanana Hurtershad-Food Oracle Ramanana Hurtershad-Food Oracle Ramanana Hurtershad Food Oracle Ramanana Ramanana Hurtershad Food Oracle Ramanana Ramana Ramanana Ramana Raman
                                                                                                                                                                                                                                    tmp 22,000
                                    last = tmp;
                                                                                                                                 am Milita Iowers, near Salvam Ineatre

Ameeros 20 2000
          display()
                                                                                                                                                                                                                         3000
                 Struct node # 9
                                                                                                                                                                                                                         1800
                                                                                                                                                                                                             2000
                                                                                                                                                                                                                                                        3000
                                                                                                                                                                                                                        last = 2000 3000
                                of ( last == NULL)
                                                                                                                                                                                                                               tmp 23000
                          else
while (9 /= last)
                                                         Printf (" Y.d", last -> link).
```

```
SRI SAI BHAVANI XEROX
   addat beg ()
                                         H.No. 7-1-209/1, Ground Floor,
                                        Ram Mirra Towers, Near Satyam Theatre,
                                          Ameerpet, Hyderabad-500 016
        struct node * tmp;
  tmp = (struct node *) malloc (size of (struct node))
      tmp -> info = num.
     tomp -> link = last -> link :
    last -> link =tmp;
                                                       5000
                                                                            4000
addafter (int pes, int num)
                                                              20 3000
                                         4000
                                                   1000
                                                                         3000
                 node * tmp, *9;
                                                      5000 tmp = $000
                                                           last = 3000
       9= qlast -> link,
     for ( i=0; i < pos-1; i++)
                                                                  Pas=2
                                                             9 = 4,000 1000
                                                             tmp = 5000
          922 -> link:
           of (9 == last -> link)
                         There are less no of nodes ");
      top = (struct node *) malloc (size of (struct node));
         tmp ->info = num.
          tmp -> link = last -> link;
           last -> link = tmp;
                                                     -> link = tmp;
          ("f(g == Last)
                             SRISAI BHAVANI YEEROX
               last=tmp;
                               H.No. 7-1-2091 Noar calvam ringali
                               H.No. 7-1-209/1, Ground Floar, 18 .
Ram Mira Towers, Near Sayan Theata
                                 ammira lowers, near Sayam Ingare
Ameerpet, Hyderabad-500 018
```

del (int num) } Struct node * tmp, *9; if (last -> link -> into = = num) / delete at beg +/ tmp = last -> link. last -> link = tmp -> link. 3000 SRI SAI BHAVANI XEROX free (tmp); RISAI BHAVAIR Ground Floor, trup = 1000

H.No. 7-1-209/1, Ground Thegire,

Towners, Near Satyam Thegire, H.No. 7-1-20911, Ground Flourium = 100

H.No. 7-1-20911, Ground Flourium = 100

Ram Mirra Towers, Near Salyam Theatre,

Ram Mirra Towers, Near Salyam Theatre,

Amserpet, Hyderabad-500 016 / = 3000 9 = last -> link; while (9! = last) { cf(q->link -> info = = data 1* delete in between */ del > 20 10 1060 1000 9-> link = tmp -> link. 9=1000 tmp. free (Amp); last-23000 2000 free (2000) /* delete last node */ 2009 tmp = last; 1000 last = tmp -> link -> link last = 3,000 2000 last -> link = tmp -> link; tmp 23000 free (tmp); free (3000) BRISAI BHAVANI XEROX HINO. 7-1-20911. Ground Floor. H.No. 7-1-20911. Ground Floor. H.No. 7-1-20911. Ground Floor. Ram Mire Towers, Near Salyam Theatre. Ram Mire Towers, Near Salyam D16 am wure rowers, wear saryam roans Africarbet, Hyderabad-500 016

```
SRI SAI BHAVANI XEROX
17/12/2018
                                      H.No. 7-1-209/1, Ground Floor,
     del (int num)
                                     Ram Mirra Towers, Near Satyam Theatre
                                      Ameerpet, Hyderabad-500 016
      2 Struct rode * tmp, * q
       if (last -> link = = last & last -> info == num) ( only one
                                                                 element deleted
         tmp = last;
           last = NULL
           free (trip); neturn;
                                         SRI SAI BHAVANI XERUA
       9= last -> link:
                                           H.No. 7-1-209/1, Ground Floor,
                                           Ram Mitra Towers, Near Salyam Theatre,
Ameerpet, Hyderabad-500 016
       if (9->link==num)
            last -> link = 9 -> link:
            free (tmp);
       while ( 9 -> link != last-)
              if (9-slink -> info == num) // element
                                                              in between
                     link = tmp -> link.
                                     SRI SAI BHAVANI XEROX
                 tree (tmp).
                                       H.No. 7-1-20911, Ground Floor,
                                       Ram Mirra Towers, Near Salyam Thealre,
                                        Ameerpet, Hyderabad-500 016
                                              11 last element deleted
                                                              9-> link = last
                free (tmp); last 29;
            Printf ("Element V.d not found In", num).
```

```
stack operations
Implement
                                          using linked list : y
   #include (stdio.h)
   # include ( conio. h)
  # define ISEMPTY top == NULL
 Struct Stack
   Ent data;
                          SRI SAI BHAVANI XEROX
                             H.No. 7-1-209/1, Ground Floor,
  Struct stack *nent;
                            Ram Mirra Towers, Near Salyam Theatre,
                             Ameerpet, Hyderabad-500 016
                                                           10
typeded struct stack node.
                                                                         2000 = 90th
                                                               1000
node + top.
Void push (int item)
                                                                        2000
                                                                  6001
     node #temp "
                                                             10
                                                                        1000
 -temp = (node *) malloc(cize of (node)).
                                                                 top=1000 2000
  temp -> data = item;
   temp > nent = top.
                                                             30
                                                                  2000
                                                                        3000
     top = temp.
                                                             20
                                                                  6001
                                                                       2000
                          SRI SAI BHAVANI XEROX
                                                             10
                                                                        1000
                           H.No. 7-1-209/11 Cround Floor
 int pop ()
                                                                  top=100020x0
                           H.NO. 1-7-ZUYIT, Ground Ploor, Theatre, Near Satyam Theatre, Ram Mirra Towers, Near Satyam of A.
                             am wirra iowers, near baryam i nearte
Marabad-500 016
Ameerpet, Hyderabad-500 016
   E node * temp;
     'ent etem ,
     if (2SEMPTY)
       E printf ("In stack is empty");
     dump = top.
      item = temp -> data;
      top = top -> nent;
       free (temp).
      return item.
```

```
Ent Peck ()
        Ef (PSEMPTY)
            print (" stack is empty
           neturn (top->data).
                                     H.NO. Tallor Nor Callon Thomas
                                     H.NO. 1-7-2041, Ground Flour,
Ram Mira Towers, Near shad-ann nia
Ram Mira Towers, Near shad-ann nia
                                       am mina lowers, near sayam ineans
Ameerpet, Hyderabad-500 016
 ent length ()
      Ent size = 0;
     for (aztop; al=NULL; aza->nent)
           Size++;
        return (size);
Void display
     of (DSEMPTY)
        Printf ("In No istem to displa
       else
         Printf ("In The data
           for (a=top; alaNULI
main ()
        choice, item, i:
    drsur();
    while (1)
         printf ("In In 1. Push");
        pront ("In 2. pop").
```

```
printf ("In 3, Peck").
 printf ("In 4. isempty").
 printf ("In S. Display data");
 printf (u/n & Length 1);
  printf ("In F. Enit");
 printf (" Enter the choice:"):
                                          SRI SAI BHAVANI XEROX
                                           H.No. 7-1-209/1, Ground Floor,
  Scanf (" ", d", & choice).
                                          Ram Mirra Towers, Near Satyam Theatre,
                                           Ameerpet, Hyderabad-500 016
 Switch (choice)
 Case 1: printf ("Enter the item to push:").
           scanf (" /.d", & item).
           push (item).
           break.
 care 2: if ((2= pop ())!=1)
           printf (" The deleted istem is: ",d", i);
           break ;
 (ale 3: if ((1=plek()) !=-1)
         Printf ("Topmost element is: ",d", i);
          break;
 case 4: if (25EMPTY)
           Printf (" In The stack is empty");
           printf ("In The stack is not empty");
           break .
          display ();
 case s:
           break;
cau 6: printf ("In The size is: /d", Length ());
          break
case 7: enit(0)}.
          default: printf (" In wrong choice ").
  1 // switch
} /while
                               SRI SAI BHAVANI XEROX
```

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Salyam Theatre, Ameerpet, Hyderabad-500 016

TREES

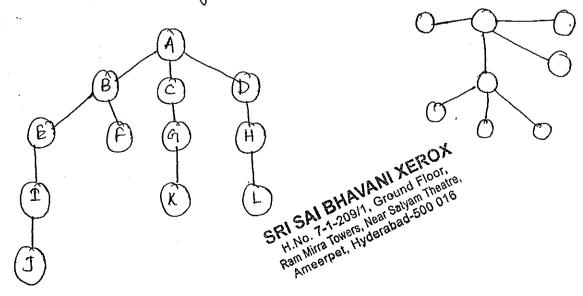
- -> The need of non-linear data-structures, improve performance of an application by reducing the execution tine.
- -> prevent loss of clata
- is improved on I wastage of memory space performance.
- -> The non-linear data Structures are (1) Trees (ii) Graphe:

Deyn of Tree:-

A Tree structure way of representing the hiearchial nature of a structure in a graphical form.

It is defined as a tree is a finite set of nodes such that there is a specially designed node called The remaining nodes are partitioned into different disjoints (Set ti, ta, tn

=> A true can be called as connected acyclic graph. -> 9t i's a hiearchical graph.



- -> A node is nothing but collection of information along with its branches.
- -> The above true has a root node of 'A' and has the following terms
 - → Degree of a node
 - -> Terminal non-terminal nodes
 - -> parent Vs chireld
 - → Siblings
 - -> Level of the node
 - -> Degree of the true.
 - -> Height of the true or depth of the true.

Degree of a node:>

- → The number of sub-trees are chiefdrens are called as degree of a node.
- → The degree of it in the above tree (eg: In above fig degree → The degree of tree is 3.

Terminal 9 non-Terminal nodes:

- → A node with no children is called out terminal nodes.

 · Cleaf nodes) i.e whose degree is equivalent to xero.

 enemples: J, F, K, L nodes in above diagram.
- → A node is said to be non-terminal whose degree is greater than xero.

 enample: B, C, D, E, G, H, I.

mample :- BICIDIE)

Parent Vs child:

 \rightarrow 9f there is a branch from $U \rightarrow V$ then , U is called as Parent and V is called as child of U.

n:- B,C,D E,F SRI SAI BHAVANI XEROX

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor,

H.No. 7-1-209/1, Ground Floor,

Ground Floor,

Holor Salyam Theatre,

Ram Mirra Towers, Near Salyam Theatre,

Ram Mirra Towers, Hyderabad-500 016

Ameerpet, Hyderabad-500 016

SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor,

Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

Level of the node :-

-) The Level of the node indecates level no of the parent plus one.

level of A = 0+1 = 1. level of B, C, D = 1+1=2

Degree of the true;

-> It is defined as the manimum degree in the tree. ex:- In above tig. degree -> 3.

Height or depth of the tree:

I the height of the tree is the largest level no in that

> In the above tree the height is 5.

-> Basically trees one two types:-

(ii) Binary trees.

-> In binary trees

Binary Trees Ramine m- way binary Trees

Binery search Enpression Balanced tree Trees (AVI) mueti-way)
Height weight
Balanced Balanced.

7 Difference Between Trees and Binary trees :-

A binary tree can be empty where as a tree cannot eg: - passible in bi

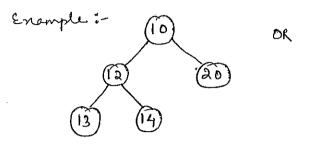
Ey Each element in binary tree has enactly two substrees cone or both of these substrees may empty), Each element in a free can have any number of substrees.

The sub tree of each element in a binary tree have one ordered, left and right sub trees.

The sub-trees in a tree one unordered.

Strictly Binary Tree :-

95 the out degree of every node in a true is either o or 2, then the tree is said to be strictly binary true



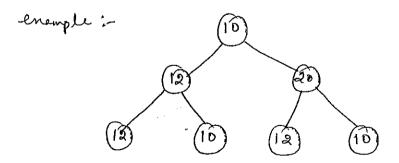
SRI SAI BHAVANI XEROX
H.No. 7-1-209/1, Ground Floor,
Ram Mirra Towers, Near Satyam Theatre,
Ameerpet, Hyderabad-500 016

Applications of Binary Trees:

- of Enpression trees are used in compilers.
- i) Huffman coding trues are used in data compression algorithm.
- -> Binary search trues which supports search, insertion and deletion on a collection of items in O (logn) (avg) etc.

Complete Binary Tree; >

In a complete binary thee, if there are 'n' nodes are enisted then all the nodes must be consecutive (sequence) (for every 2i+1 there must be enist 2i)



SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

Representation of Binary Trees:

SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

I The storage representation of Binary trees in two ways

(i) sequential Representation.

H.No. 7-1-209/1, Ground Ploot, Ram Mitra Towers, Near Safyam Theatre, Amesrpet, Hyderabad-500 016

 \bigcap

(

(

(

(i) Linked live Representation 100H punois 1/607-1-1 ON'H

SRI SAI BHAVANI XEROX

Sequential Representation:

y The sequential representation is nothing but arrays.

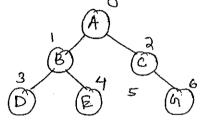
- An array can be used to store the nodes of a binary tree.

-> The nodes can be accessed sequentially

I suppose a binary tree T' of defth d' the atmost (2d-1) nodes can be there in T.

 \rightarrow . So the array size to represent a binary tree of deft depth 3. is $2^3-1=7$ so A[7]

> To perform the operations, we have to i'dentity root, left child and Right child.



(i) The root of a node having inden 'n' can be obtain by (n-1)/2.

for enemple: To find the root of B where array index n=3, then the root node index can be obtained as

(3-1)/2 = 2/2 = 1.

i.e A[1] is the root of BD which is B

SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016 (ii) The left child of a node having an index n' can be obtained by (2n+1).

Enample: To find the left child of C where array enden n=2, this can be obtained by $(2n+1)=(2\times2+1)=5$ is the left child of C which is null so as per the fig. there is no left child of C.

(iic) The right child of a node having array inden n' can be obtained by the formula (2n+2)

Enample: To find the right child of the B, where array index n=1, this can be obtained by 2n+2=2n+2=4, i.e A[4] is the right child of B which is E.

(iv) It the left child is a array orden n, i.e right child. is (n+1). Similarly if a right child of index n then its left child is (n-1).

The above tree is not good representations by using occasionays it is better to represent in complete binary trees.

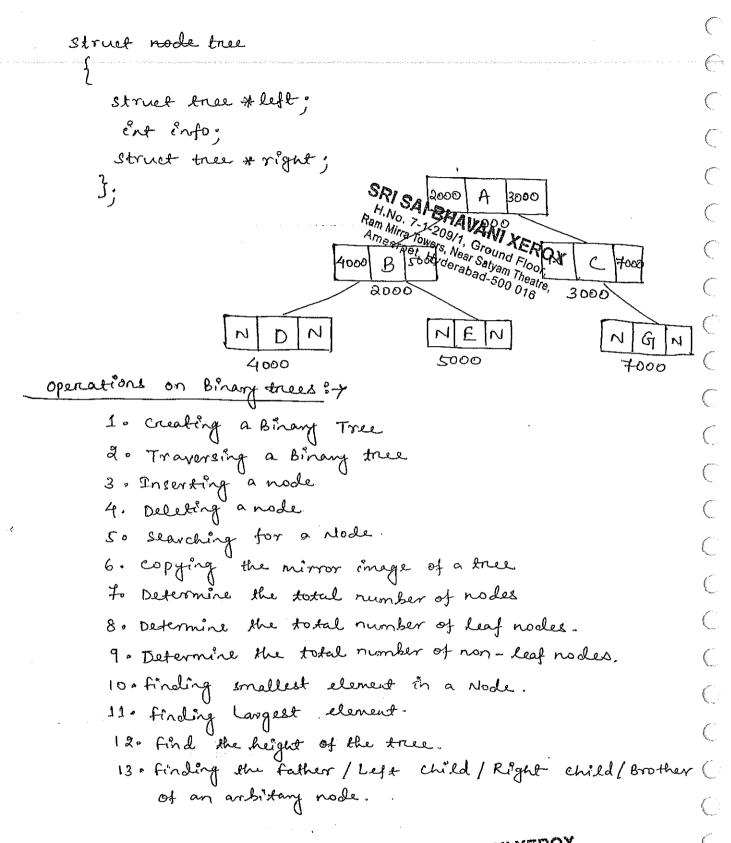
Linked List Representation:+

- The most popular way and practical way of representating a binary tree using linked list concept.
- 70 implement the trees a node will contain minimum 3
 - (i) Information field.
 (ii) previous field / left child
 - (iii) Next field / Right child.

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

prev	Photo	nent
		1
}	1 1	



SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mina Towers, Near Satyam Theatre, Ameerpet, Hyderabad-509 918

Traversing a Binary Tree: >

- Tree traversal is one of the most common operation, perform the tree datastructures, i't is a way in which each node in the tree is visited enactly only once in a systematic manner.
- They are the standard base to represent tree tourersals

Like :- > Prue-order

> In order

> poltorder

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

-> we can also have Level order.

-> This traversal techniques can be implemented in two ways

(i) Iteration

(ii) Recursion

With Recursion of prelorder it

7 To traverse a non-empty binary true in pre-order following steps one of the be processed.

1. Visit the root node

20 Traverse the left sub tree in preorder

3. Traverse the right cub tree in preorder.

7 That i's in preorder traversal, the root node i's visited (or processed) tirst, before traveling through left and right sub tree recursively.

with iteration :>

Algorithm for preorder Traversing:

SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

Step 1: Set Top -1

Stack [1] - NULL

PTR - Root

Stepa: Repeat 3-5 while PTR 1 = NULL

step 3: Apply process to info [PTR]

Step 4: is Right [PTR] |= NULL then

TOP - TOP +1

Stack [TOP] - Right [PTR]

Step 5°, if Left[PTR] = NULL then

PTR - LEFT [PTR]

else

PTR

Stack [Top] H.No. 7-1-209/1, Ground Floor,
Ram Mirra Towers, Near Satyam Theatre,
Ameerpet, Hyderabad-500 016

Inorder Traversal Recursively:

I The inorder traversal of a non-empty binary true is defined as tollows:

1. Traverse the left sub true in order

20 Visit the root node

3. Traverse the right sub tree in order

I Inorder traversal, the left sub tree is braversed recursively, before visiting the root.

-> After Visiting the noot the right sub true is traversed recursively, in order dashion.

Algorithm:

(assigning)

Step 1 ? set 70p (-1

Stack [1] - NULL

PTR - Root

SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

step 2 % Repeat while PTR! = NULL

TOP (TOP+1

Stack [TOP] (PTR

PTR (Left [PTR]

Step 3: PTR - Stack [Top]

Step 4: Repeat Steps S-7 while PTR!= NULL

Step 5 . Apply Process to Info [PTR]

step 6: if Right [PTR]]=NULL then

PTR - Right [PTR]
Cro to step 2

Step 7: PTR - Stack [TOP]

SRI SAI BHAVANI XEROX Ground Floor, Ground Theatre, Salvam Toles, Near Salvam Toles, Near Salvam Toles, Near Salvam Toles, Near Salvam Nillra Towers, Hyderabad-500 016

TOP - TOP-1

Post order Traversal :>

The post order traversal is a non-empty binary tree can be defined as:

1. Traverse the left sub true in post order

2. Traverse the right sub tree in post order.

3. Visit the root node.

In past order traversal, the left and right sub true(s) are recursively processed before Visiting the root.

Algorithm:

Step 1: Sit TOP - 1

Stack[1] - NULL

PTR - Root

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderahad, 500 2458

Step 2: Repeat 2-5 while PTR != NULL Ameerpet, Hyderabad-500 016

Step 3: TOP (TOP+1

_Stack[TOP] (PTR

Step 4: if Right [PTR] = NULL

TOP (- TOP+1

Stack [TOP] (-- Right [PTR]

Steps: PTR - Lest [PTR]

Step 6: PTR - Stack [TOP]
TOP - TOP-1

Step 7: Repeat while PTR > 0

Apply process to Info [PTR]

PTR

Stack [TOP]

TOP

TOP

TOP-1

Steps: FTR < 0
PTR ← POTR - PTR
Coo to step 2.

SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016 order

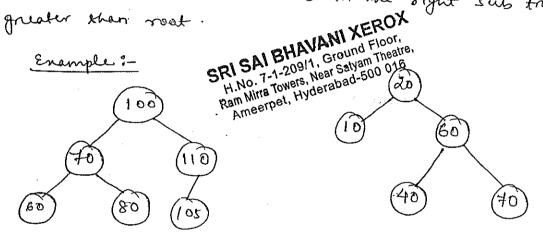
The level of townsal, visiting every node from level 1

to level n.

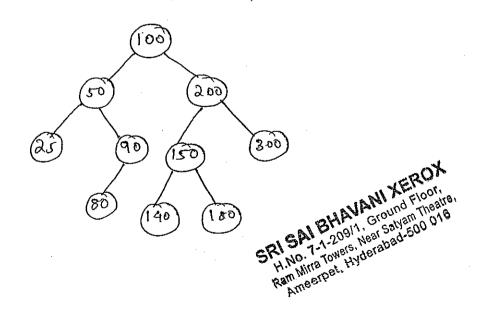
Detore implementing the traversal technique try to implement any binary trees.

Binary Bearch Tree: > (BST)

each node in the true, elements in the left sub true are less than root and elements in the right sub true are quester than root.



En: 100,50, 200, 90, 80, 25, 300, 150, 180, 140



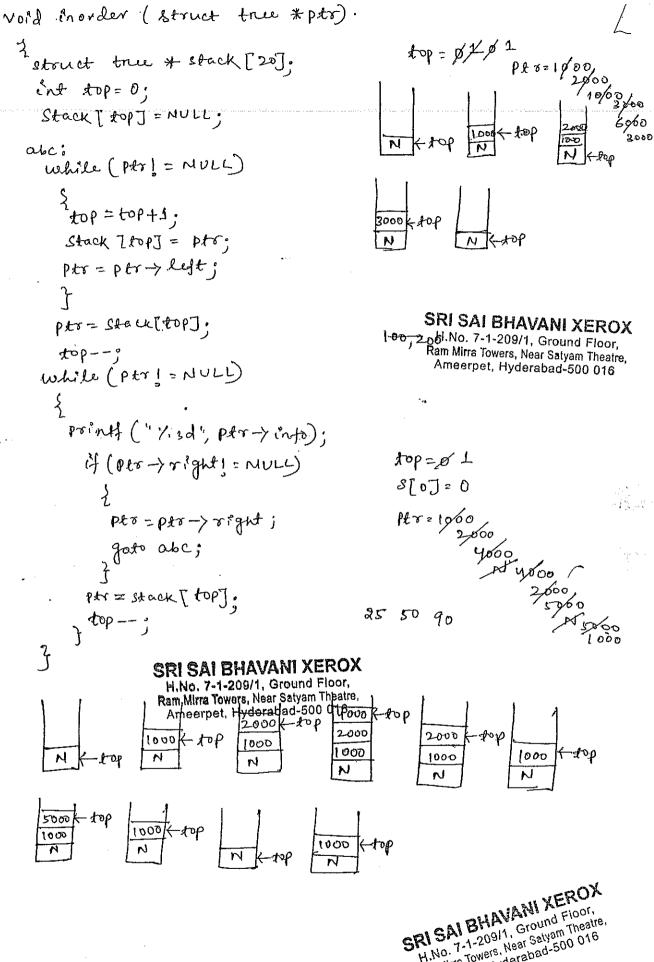
```
Diwrite a program on BST with implementing tree transmal
  techniques with ileration Concept.
Struct tree
      Struct true * left:
       Ent onto;
       Struct true * right;
     Struct true * Toot = NISAIBHAVANI XEROX
     Stouct tree of create (H.No. 7-1-209/1, Ground Floor, Near Satvam Theatre
                              Ram Mirra Towers, Near Satyam Theatre,
      Void display (struct Amagines Hyderabad-500 016
      void preorder (ctruct tree *).
      void inorder (struct tree *);
      void postorder (stout tree *);
     main()
        int ch ?
        do
         Ź
           α:
             eloscor();
            printf ("in CHOICE ACTION In In");
            print ("1. cruation of Tree m/n"):
            printf (" 2. Display of Tree In");
            print ("3, Enit \n");
            switch (ch)
                            root = create ():
                 case 1:
                             goto a;
                            display (root);
                            ·eh= o;
                                               SRI SAI BHAVANI XEROX
                                                H.NO. 7-1-209/1, Ground Thrank
                              getch();
                                                TI.IYU. 1-1-2031 I, SI SAYAM Theatre, Near Sayam Ann nia
                 case 3: enit (6);
                                                  'Ailile≱that' Hàqetspaq-200 01€
'an wing iomer≥' lagar ⊃giàsii iuesite
```

21/12/2016

```
prints ("In Enter your choice: ").
    scanf ( " y.d", gch);
   uhile (ch! = 3);
Struct tree * create()
    struct tree & head,
    ent n,x,
     chreeres;
 Print ("In Enter how ma
                                       Do you want to create :").
  Scanf ( " y.d ", &n);
                                into 1 Node : ");
 printf ("In Enter the
  scanf (" /.d ", & x) 5
  head = (stouct
                    the malloc (size of (struct tree *)).
  head -> info =x;
  head -> left = NULL;
                                                  100
  head -> right = NULL;
                                                  1000
                                                             head=1000
  for (121; i(n; i++)
   printf ("In Enter Data into %d mode i", i+1).
   scanf ( " y. d ", & x);
                                              2000
                                                     3000
   first = head;
                                                     M
  while (first ! = NULL)
                                   4000
                                     M 52 M
     prev = first ;
    引(first->info>ル)
                             N 25 N
                                           SRISAI BHAVANI XEROX
                                            H.No. 7-1-209/1, Ground Floor,
                                    N.
                                            H.140. 1-1-2031, Gradio Thealie,
Ram Mira Towers, Near Salyam na R
                                              Ameerpet, Hyderabad-500 016
    elie
        first - first -> right;
    3 Tuhole
```

```
temp = (struct tree #) malloc (size of (struct tree));
   temp -> info = u.
   temp -> left = NULL;
  temp -> right = NULL;
  if ( prev -> into >x)
      prev -> left = temp.
 else
                                      SRI SAI BHAVANI XEROX
                                       H.No. 7-1-209/1, Ground Floor.
     prev -> right = temp;
                                      Ram Mirra Towers, Near Satyam Theatre,
                                       Ameerpet, Hyderabad-500 016
  3/1/for
return head:
 Vord
         display (struct tree * pts)
     int t;
     chrocy ().
     Printf ("In In which method the tree has to traverse In In").
     printf ("1
                  PREORDER Inin");
                                          -> preorder NLR
     printf ("2
                  2NORDER In ").
                                          100,50,28,90,200,150,300
                 POST OR DER m");
                                          -> pastorder LRN
     printf (" Enter your choice : ");
                                            25, 90,50, 150, 300, 200, 100
     scanf (" /.d", & t);
                                         -> Inorder LNR
     Switch(t)
                                          25,50,90,100, 150,200, 300
       case f: preorder (ptr);
                 break;
                                              SRI SAI BHAVANI XEROX
       Case 2: (norder (ptr);
                                               H.No. 7-1-209/1, Ground Floor,
                                               Ram Mirra Towers, Near Salyam Theatre,
                 break
                                                Ameerpet, Hyderabad-500 016
                 post order (per);
       care 3:
                  break:
       default! printf (" (2000 ... ");
```

Void preorder (struct tree * ptr) int top = 0; struct tree & first = ptr, * stack [20]. Stack[top] = NULL; top=&XXXXXX vehile (first ! = NULL) S[0] = N/SIL3000 print ("1.3d", first->into): if (time-) right; = NULL) Ptr21000 5000 first = 1000 5213=3000 なのナナ・ 001= E2] 2 ST1] = 7000 **বিচচ**চ stack [top] = first -> right; 34 1 2 100 1 hoer that the are not so the state of the st if (first -> left ! = NULL) first = forst > left; else first = Stack[200]: top -- ; t op = 1 1 S[o] = NULL. P&r=1000 first = 1000 2000 4000 Void Inorder (struct 4600 50603,400 6,600 4960 5000 K-top 3000 3000 3000 SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, H.NO. 1-1-2091, Graniu Theatre, Near Salyam Theatre, Ram Mirra Towers, Near Salyam Oth _{am mirra lowers, Near Saryam i neare Ameerpet, Hyderabad-500 016}



H.No. 7.1-20911, Ground Floor, Ram Mirra Towers, Near Salyam Theatre, Ameerpet, Hyderabad-500 016

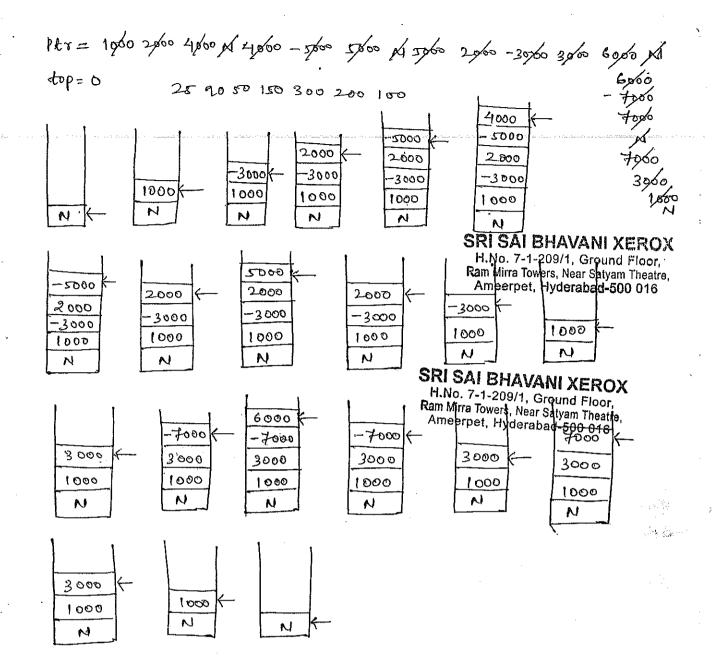
```
Wold postorder (struct true * ptr)
     int top = 0.
     Struct tree & stack [20].
       Etack[top] = NULL;
    while (pto ! = NULL)
         top++;
        Stack [top] = ptr;
         if ( per -> right ) = NULL John Miles
              top++;
             Stack [topas] =- (ptr -> right);
          Ptr=ptr -> left;
         Ptr = Stack[top]
        while (ptr >0)
           printf (">d", ptr->info);
            Ptrz stack[top];
              top - - :
         if (per(0)
             ptr=-ptr
             godo asc.
```

```
Void postorder (Struct free A
 long int stack [20], pointer,
 Struct tree & ptr;
  int top = 0.
   per = roat.
   Stack[top]=0;
      Mile (Perj=NULL)
       如十十;
      Stack[top]=(int) ptr:
     i'f (per->right | = NULL)
          10P++:
     Stack[top] = - (int)
                       (pts->right).
    Pto = Pto -> left.
     top -- .
    while (pointer; o)
   ptr = (stout toee *) pointer.
    Printf ("Y.d 12", ptr->info).
    pointer - stack[top].
    if (pointer (o)
      Pointer = pointer.

r= ('struct tree *) pointer;

goto ab (;
```

SRI SAI BHAVANI XEROX SRI SAI BHAVANI XEROX H.NO. 7-1-20911, Ground Floor, H.NO. 7-1-20911, Ground Theatre, Ram Mirra Towers, Near Satyam Theatre, Ram Mirra Towers, Near Satyam 016 Ameerpet, Hyderabad-500 016



SRI SAI BHAVANI XEROX H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

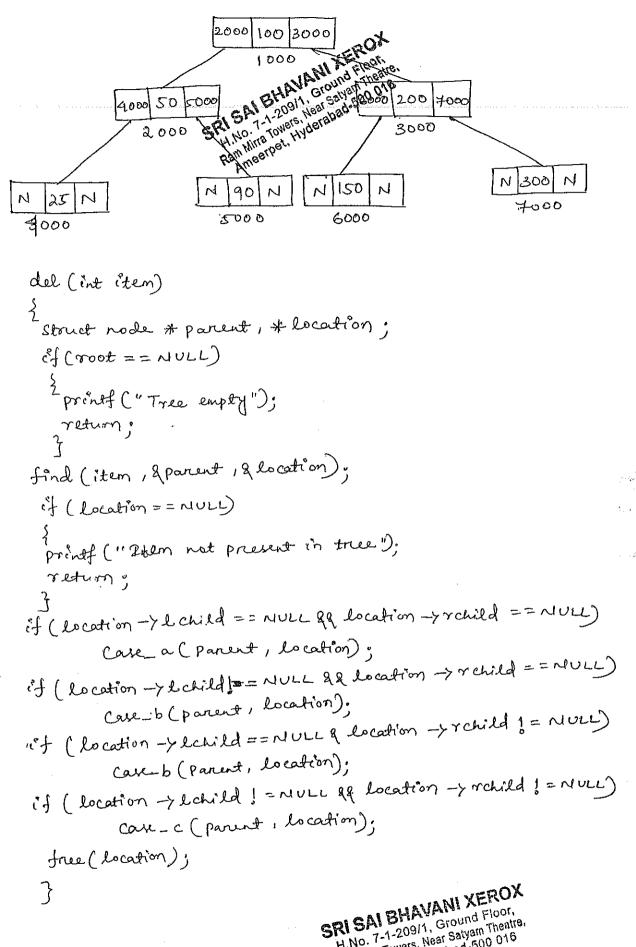
```
23/12/2016
Q) Implement the BST
                                     of creation or insertion, find, Delete
     operations.
        # include (stdio.h)
        # include (malloc.h)
        struct node
           struct node * lchild;
           struct node * rchield:
                                   SRI SAI BHAVANI XEROX
          f * root:
                                     H.No. 7-1-209/1, Ground Floor,
                                    m.No. 1-1-2091, Ground right, Near Salyam Theatre, Near Salyam Theatre, Near Salyam Theatre, Near Salyam Theatre, Hyderabad-500 016
Ram Mirra Towers, Hyderabad-500 016
Ameerpet, Hyderabad-500
       main()
        int choice, num
        root = NULL ;
       while (1)
         print ("\n");
         print ("1. Insert m")
        print (" 2. Delete \n");
         printf (" 3. Inorder Traversel In").
         pronof (" 4. Display (n");
         printf ("5. Quit in");
         printf ("Enter your choice : ");
         Scanf (" "d", & choice):
         Switch (choice)
                      printf ("Enter the number to be interted: ");
                       scarf ( " /d ", & num) =
                       Insert (num).
                       break:
           Case 2: printf (" Enter the number to be deleted");
                       scanf (" /d", &num);
                                             SRI SAI BHAVANI XEROX
                       del (num).
                                              H.No. 7-1-209/1, Ground Floor,
                                              Ram Mirra Towers, Near Salyam Theatre,
                        break,
                                               Ameerpet, Hyderabad-500 016
```

```
break;
                display (root, 1);
  Case 4:
                enit(0);
  Case 5:
                                                                  SRI SAI BHAVANI XEROX
                                                                   H.No. 7-1-209/1, Ground Floor,
         default:
                                                                  Ram Mirra Towers, Near Satyam Theatre,
                      printf ("wrong choice \n");
                                                                   Ameerpet, Hyderabad-500 016
find (int item, struct node **par, struct node *loc)
    Struct node *ptr, *ptrsave;
        Ef(root == NULL) // tree empty
            *loc = NULL;
            * par = NULL;
              return;
        if ( item== root - y item )
                                                      is at root.
                                          11 item
            * loc = root .
           * par = NULL,
               return;
    Minitialize ptr and ptreave
                                                  SRI SAI BHAVANI XEROX
        if (item ( root ->info)
                                                   H.No. 7-1-209/1, Ground Floor,
                                                   H.NO. 1-7-20917, Ground Floor, Theatre, Near Salyam Theatre, Near Salyam Theatre, Near Mari Mirra Towers, Near Bad-500 016
Ram Mirra Towers, Hyderabad-500 016
Africarpet, Hyderabad-500
            Ptr = root -y lchild;
             else
              per = root -> rehild;
              ptrsave = root;
         while (ptr = NULL)
               if ( item == ptr -> into )
                    + loc = Petr.
                    * par = pto save;
```

inorder (root).

3:

```
SRI SAI BHAVANI YEROX
                                             H.NO. 7-1-209/1, Ground Floor,
           ptreave = ptr:
                                             Ram Mirra Towers, Near Bay And Base,
                                              Ameerpet, Hyderabad egg gi g
            if ( item < ptr-> info)
               ptr=ptr-> lchild:
                ptr=ptr->rchild;
         . ] // while.
            *loc = NULL; / item not found
            *par = persave.
24/14/2016
       insert (int item)
        Ensert node * 1mp, * parent, * location;
      find (item, sparent, slocation).
         if ( Location ! = NULL)
          printf ("Item already enisted");
          return;
       tmp = (struct node *) malloc (size of (struct node));
         trup -> info = item:
         trop -> echild = NULL;
         trop -> rehild = NULL.
                                              SRI SAI BHAVANI XEROX
                                                H.No. 7-1-209/1, Ground Floor.
          Ef ( parent == NULL)
                                               Ram Mirra Towers, Near Satyam Theatre,
                                                 Ameerpet, Hyderabad-500 016
            root = tup,
       else
           'if (iten < parent -> info)
                parent -> lchild = tmp;
                 parent -> r child = tmp;
```



H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Thentre, Ameerpet, Hyderabad-500 016

```
call_a(struct node * per, struct node * loc) 1/2000
 { if (par == NUL) // item to be deleted is root node
      root=NULL:
                                                       200100 3000
   عيله
       if (loc == par -> lchild)
                                              N 50 5000
                                                                       210
                                               2000
          par -> echild = NULL ;
           par -> rehild = NULL;
   > It ease_a
Case b (Struct mode * par, struct node * loc) //1000
  } struct node * child;
                                         SRI SAI BHAVANI XEROX
                                          H.No. 7-1-209/1, Ground Floor,
    / initialize child
                                         Ram Mirra Towers, Near Satyam Theatre,
                                          Ameerpet, Hyderabad-500 016
  of (loc->lchild=NULL)
    child = loc -> lchild;
     child = loe > r child; 115000
    if (par == NULL) // item to be deleted is root node
      root = child;
   else
     if (loez= par -> likild)
      par -> lchild = child;
      par -> rehild = child;
    4 // case_b
  care_c (struct node * par, struct node * loc) //1000 3000
   Struct node * ptr, * ptreave, * suc, * parsuc.
   I find inorder successor and its parent
       ptrsave = loc; //3000
       Ptr = loc -> lohild ; 1/6000
                                             SRISAI BHAVANI XEROX
      while ( ptr->lchild }=NULL)
                                               H.No. 7-1-20911.
                                               M.NO. 1-1-2001, Orania Theate, Near Salvam Theate, Near Salvam Theate, Near Salvam 1016
                                                <sub>anı mırıs</sub> rowers, wear sayam rneare
Arrestpet, Hyderabad-500 016
Arrestpet,
         1 ptrsove = ptr;
           ptr = ptr - 7 lchild;
```

```
Suc=ptr; 1/6000
  parsue = persave; // 3000
  if ( suc-> lohild == NULL && suc-> rohild == NULL)
        Care a (par euc, suc) , // 3000 6000
     else
        Case -b (parsue, sue).
if (par == NULL) // Ef item to be deleted is root node
   root = suc.
 عداد
    if ( loe = = par -> lchild)
                                                 SRI SAI BHAVANI XEROX
                                                  H.No. 7-1-209/1, Ground Floor,
    par -> lchild = Suc.
                                                 Ram Mirra Towers, Near Satyam Theatre,
                                                   Ameerpet, Hyderabad-500 016
     par -> rchild = suc:
  Suc -> lchild = loc -> lchild.
  Suc -> rehild = loc -> rehild.
( 1/ case_c
  inorder (struct node * ptr)
   }
if (noot = = NULL)
        printf ("Tree is empty");
     ( ptr ! = NULL)
        & inorder (ptr-y lchild),
        printf (" %d", ptr > info);
         inorder (ptr-ynchild);
                                          SRI SAI BHAVANI XEROX
     3 // inorder.
                                            HIND, 7-1-209/11, Cannon Thomas
                                            H.NO. 1-1-2091, Ground Flouris
Ram Mirra Towers, Near Satyam Theatre,
                                             <sup>श्चम लातव</sup> Iowers, Near Sayam I neare
भूग लातव Iowers, Near Sayam I neare
```

```
display (struct node * ptr: int level)

int i,

if (ptr: = NULL)

display (ptr->rchild: level + i);

printf (" 'n');

for(i=0; i< level; i+t)

printf (" '');

display ("ptr->lchield: level + i);

if display.
```

31/12/2012

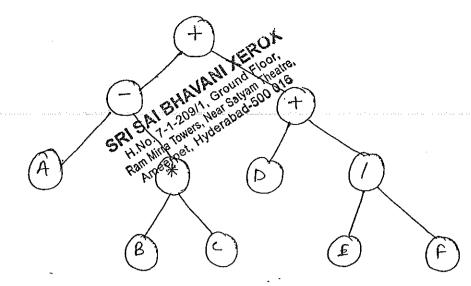
Expression Trees:

- expression tree for an arithmetic relational or logical expressions can store in an expression trees.
- The enpression trees also from the family of binary trees
- I when an enpression tree is forming we need to follow the rules, the paranthesis in the enpressions donat appear.
- The leaves of the variables for the constant in the expression.
- -expression.

 Leaf

 The non-beaver nodes are the operators in the expression that.

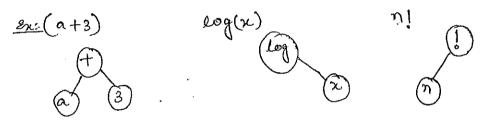
- > A node for binary operators has two non empty
- -> A node for unery operator has non-empty cub-since.
- In the given enpression, the operators with the least priority will acts a root mode (according to descending).



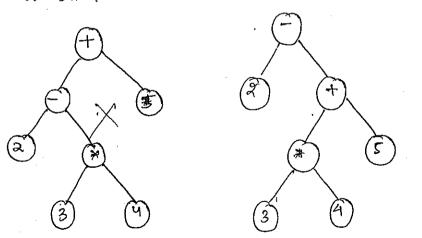
Infin Motation: ((A-B*c) + (D+E/F)).

in Such a way that an inorder traversal produces the original enpression without paranthesis.

I The least priority of operator will acts as a root node and followed by other operator.

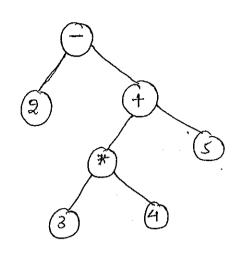


2-3+4+5

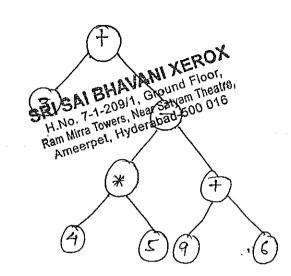


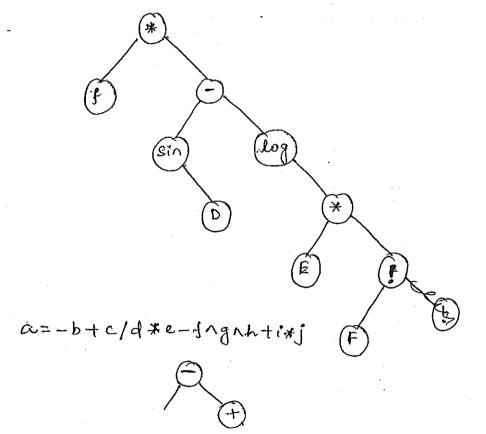
SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Salyam Theatre, Ameerpet, Hyderabad-500 016



f(A, B,C) * (sin(D) - Log(E*F!))





SRI SAI BHAVANI XEROX
H.No. 7-1-209/1, Ground Floor,
Ram Mirra Towers, Near Satyam Theatre,
Ameerpet, Hyderabad-500 016

why enpression trees required?

y for evaluations of enpression, generating compiler code to compute the enprussion value and enecution

-> Basically all the compilers will convert into expression at the time of evaluation.

symbolic mathematical operations on the enpressions 2/01/2017

1. Implementation of an enpression true to perform

traversals.

AIM 3-

To implement an enpression tree and to perform in-order and post-order traversals.

Algorithm :-

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre. Ameerpet, Hyderabad-500 016

Step1: - Start the process

Step 2 %- Pritialize and declare variables

step 36- Enter the pushfix expression that can be stored in the stack.

Step43- In pop operation, check the top of the stack is empty; otherwise stack[top] = Node, and decrement the top value.

Step 5:- In push operation, check the size of the stack, stack is not full, top is incremented and stack [top]=Node

step 6: - Allocate the memory for new character and our igned left and right pointer is NULL.

step for If character value is +, *, /, -, pop the right and left pointer to temp and push into stuck

step 8: - The pre-order function is traverse the most node and left, right node of the tree.

SRI SAI BHAVANI XEROX

H.No. 7-1-209/1, Ground Floor, Ram Mirra Towers, Near Satyam Theatre, Ameerpet, Hyderabad-500 016

```
# include (stdio.h)
  # include (conio.h)
 # include (alloc.h)
# include (ctype .h)
  # define Size 20
                                                                                                                               SRI SAI BHAVANI XEROX
typedet struct node
                                                                                                                                     H.No. 7-1-209/1, Ground Floor,
                                                                                                                                    Ram Mirra Towers, Near Satyani Theatre
                                                                                                                                         Ameerpet, Hyderabad-500 016
       chan data;
        Struct node * left;
        struct node * right;
        } berne;
         btree # start [ size];
                int top .
         Void main ()
                                                                                                                                                                      SRI SAI BHAVANI XERON
                  bence & root;
                                                                                                                                                                          H.No. 7-1-20911, Ground Fleer, H.No. 7-1-20911, Ground Fleer, H.No. 7-1-20911, Near Salvario, Front Plant H.No. 7-1-20918, Near Salvario, Front H.No. 7-1-20918,
                   Char enp [80];
                                                                                                                                                                                am mma iowers, near payam ineght
hyderabad-500 016
Ameerpat, Hyderabad-500
                   btree * create (char emp[80]).
                   Void preorder (blue + root);
                     Chroceco;
                     Printf ("Enter the postfix enpression");
                     sconf ("/s", eup).
                      top = -1:
                        root = create (emp);
                         print ("In The tree is created");
                          printf ("In The preorder traversal of tree In");
                          preorder (root);
                          getch();
```