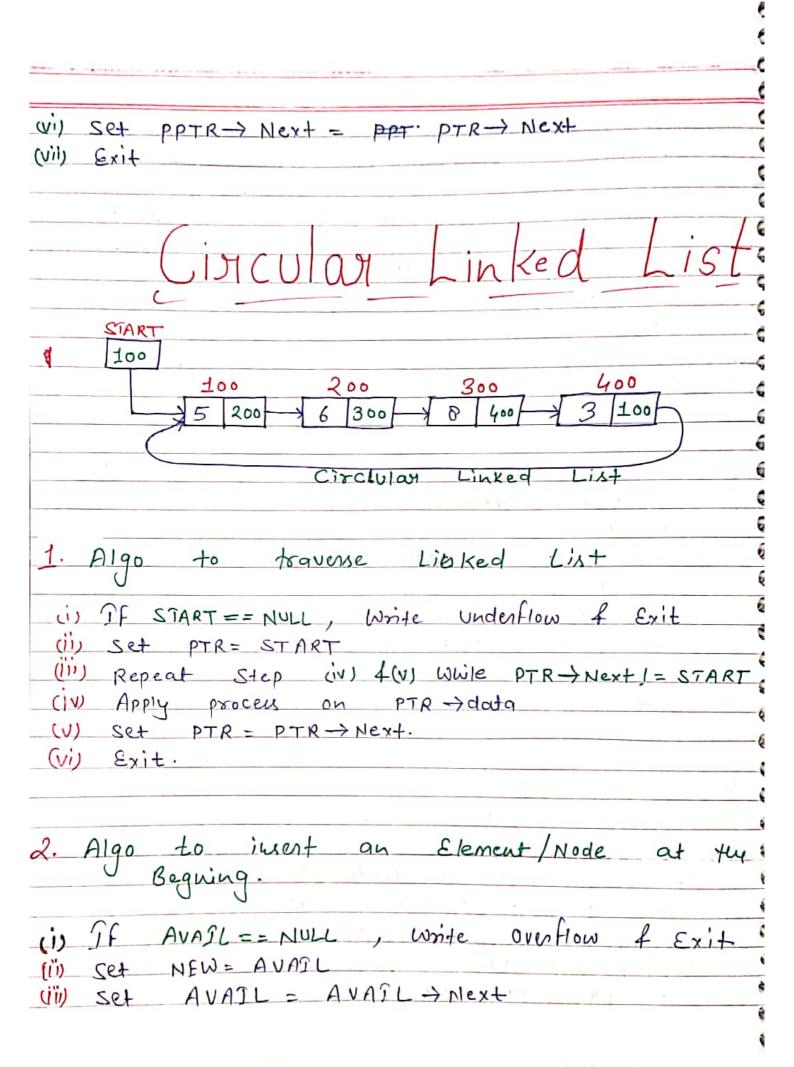
Linked hist,
Pala Made Data Next Next Node Node
Singly Linked List,
I. Algo to traverse a Linked List. Node contain Add of First Node START Node Add of Node
10 200 300 400 X NULL
(i) Set PTR = START (ii) Repeat S-lep (iii) and (iv), while PTR! = NULL (iii) Apply process of PTR - data. (iv) Set PTR = PTR -> Next (v) Exit
2. Algo to insert an Element in the beginning
AVAIL -> AVAIL is a pointer which contains /points the free AA Blocks of memory.
AVAIL is the Linked List of free memory 1310cks.

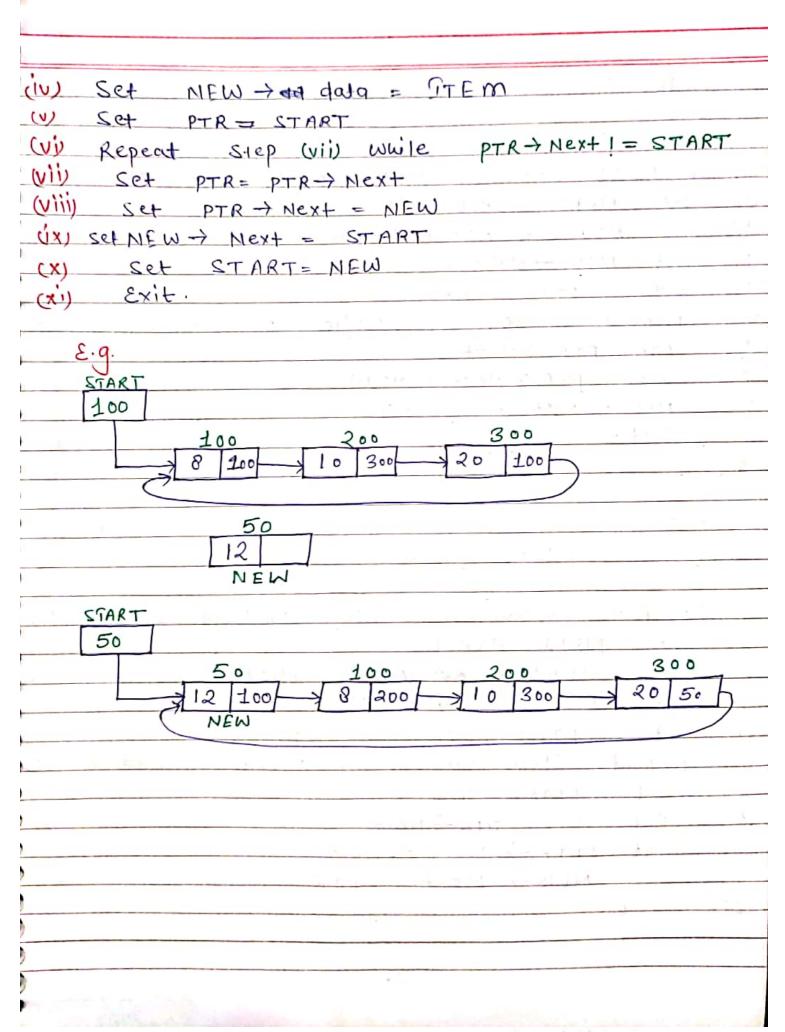
Checking Available Space in memory. , Write overflow AVAIL == NULL Set NEW = AVAIL AVAIL = AVAIL -> Next // Switching Set AUAIL Next Block For later (iii) Set NeW - Next = START (iv) Set START = NEW (V) Set NEW -> data = TTEM (Vi) Exit Which inserted is element in New C.9. START 100 100 NEW-> data = 2 100 NEW > Nex+ = START 4 Address of NEW Node START 50 300 400 100 200 300 -10 2. 2. 2 2

```
Algo to Insert an Element after the
     particular location. by value.
 (i)
       AVAIL == NULL, write overflow & Exit.
 (ii)
     Set NEW= AVAIL
 (iii)
     Set AVAIL = AVAIL -> Next
(iv)
     Set NEW -> dalg = JTEM
     SC+ PTR = START
(N)
     Repeat Step (vii) while PTR -> data 1= VALUE
(Vi)
(vii)
      Set PTR = PTR -> Next
(Viii)
     Set NEW-> Next = PTR-> Next
(ix)
     Set PTR -> Next = NEW
) (x) Exit
          Algo to insert an Element before
         particular location
                          by value.
     START
      100
                                       130
            100
                     110
                              120
             110
                               130
    Ingenting Element before third
                                    Mode
                                           WW oh
          Value / daya
       AVAIL == NULL, write overflow & Exit
         NEW = AVAIL
2 (1111) Set AURIL = AURIL -> Next
Div set NEW -> data = ITEM
VI SEL PTR= START
 (VI) Set PPTR = PTR
```

```
Vij Repeat Step (viii) and (ix) while pTR+datq!= 4
(1) Set PPTR = PTR
 (x) Set PTR= PTR-> Next
(X) Set PPTR -> NEXT = NEW
 (Xi) SET NEW -> NEXT = PPTR
tix3 (iix)
 5. Algo to Inest Node At the End.
i) If AVAIL == NULL, write overflow & Exit.
 (ii) Set NEW= AVAIL
 (ii) Set AVAIL = AVAIL -> Next
 (IV) Set PTR= START
 (V) Repeat Step (vi) While PTR! = NULL
 (UI) PTR= PTR -> Next
 (vii) PTR -> Next = NEW
 (Viii) NEW-> Next = NULL
ix) Exit
     START
      100
                               updated value
           100
                             NEW
```

```
6. Algo to delete first Node of Linked
    IF START == NULL, write underflow & Exit
 ci
 (i) Set PTR = START
 (iii) Set START = PTR -> Next
 (iv) Exit Free CPTR)
 7. Algo to delete last Node of Linked
) U) If START == NULL, write underflow & Exit.
(ii) Set PTR = START
(iii) SET PPTR=START
D(iv) Repeat Step (v)foi, wwile PTR -> Next ] = NULL.
(V) Set PPTR= PTR
    Set PTR = PTR -> Next
o (i)
 (Vii) Set PPTR -> Next = NULL
 (vili) free (PTR)
 (ix) Exit
2
8. Algo to delete an 9 Node from Linked
    List by value / dalg.
i) If START == NULL, write underflow & Exit.
(ii) Set PPTR = PTR = START
 (iii) Repeat Step (iv) & (v) While pTR > data! = VALUE
(iv) Set PPTR = PTR
 (V) Set PTR= PTR -> Next
```





	2
3. Algo to Invert Element at the End :	3
of Linked List.	-3
	3
ci) If AVAIL == NULL, write overflow of Exit	3
(ii) Set NEW= AVAIL	3
(li) Set AVAIL = AVAIL -> Next	3
(IV) Set New -> dada = ITEM.	2
(V) Set PTR= START	2
Depeat Step (vii) Wille DTD Novt 1 - START	3
PTR = PTR -> Nex+	2
(viii) Set PTR-> NEXT = NEW	3
(ix) NEW -> NEXT = START	3
(X) Exit.	5
of .	-5
4. Algo to lisent an Element/Node before	
4. Algo to Susent an Element/Node before a particular location by data/value.	
	2
i) If AVAIL == NULL, write overflow & Exit	
(ii) Set NEW= AVAIL	
(ii) set AVAIL = AD AVAIL -> Next	
(IV) Set NEW -> data = ITEM	5
(V) Set PTR = PPTR = START	0
(Vi) Repeat Step (Vii) & (Viii) While PTR->data1=VAL	
(VI) Set PPTR = PTR	-
(VIII) Set PTR = PTR -> Next	-
(IX) Set PPTR -> NEXT = NEW	6
(X) Set NEW -> Next = PTR	
(Xi) Exit	2

```
5. Algo to Smort a Node After the
 particular location by value / data.
(i) If AVAIL == NULL, write overflow & Exit.
(ii) Set NEW= AVAIL
(iii) Set AVAIL = AVAIL -> Next
(iv) Set NEW -> day = ITEM.
(V) Set PTR = START

(Vi) Repeat Step (Vii) While PTR > clay + PTR > data) = VAL
(Vii) Set PTR=PTR -> Next
(Viii) Set NEW > Next = PTR -> Next
in Set PTR -> Next = NEW
(x) Exit
6. Algo to delete first Node of Linked
i) IF START == NULL, Write overflow & Exit
(1) Set PPTR = PTR = START
(11) Repeat Step (iv) While PTR=> Next 1= START
(IV) Set PTR = PTR -> Next
(V) Set PTR-) Next = START-> Next
(VI) SEX START = START -> Next
(Vii) Free ( PPTR)
(viii) Exit
```

. Algo to delete Linked list.	Last	Node	oF	Circular
If START == NULL Set PPIR:PTR = STA	, Write	overflow	4	exit.
) Reapent Step (in) Set PPTR = PTR	1) f(v) while	PTR	→ Nex	t I = START
Set PTR= PTR- Set PPTR-> Ne	> Nex+			Le Le
i) free (PTR) ii) Exit	<u></u>		7 1	l.
		L . 1		L g'
Algo to Delet	te 9 r	vode	from	Circulay
Linked List	64	data/ Val	110.	L. J.
	J			
Same As	Singular			List.
				List

Double Linked List
Two way Linked List
prev Info Next
-> The Information field Info stores the data of a Node.
The pointer field Next Contains the Address
The pointer field preu Contains du Addres
1. Algo To traverse double Linked List
100 START 100 START 100 200 300 X 8 200 100 12 300 200 13 X
(i) Set PTR= START (ii) Repeat Steap (iii) and (iv) While PTR 1= NULL (iv) Apply process on PTR -> dala (iv) Set PTR= PTR-> Next (v) Exit

2. Algo to Thiest an Element in beginning.	the
beging.	
U	
(i) If AVASL == NULL, write Overflow & Ex	it
(i) Set NEW = AVAIL	
(iii) Set AVASL = AVASL - Next	- 1
(iv) Set NEW-) days = STEM	
(V) Set NEW-) Next = START	, 1
(Vi) Set NEW -> prev = NULL	
(VI) Set START -> prev = NEW	1
(VIII) SET START = NEW	
lix) Exit.	
	,
	//
3. Algo to Truest an Element At W	4 5
3. Algo to Turent an Element At Y	6
	6
(i) If AVAIL == NULL, Write Overflow & Exit	. 5
(ii) Set NEW= AUNIL	
(iii) Set AVAIL= AVFIL -> Next	ę
(iv) Set PTR = START	
(V) Set NEW -> dada = ITEM	6
(Vi) Repeat Step (Vii) While PTR-> Next 1	= ND// a
VI) Set PTR = PTR -> NexL	-10000
(viii) set PTR-> Next = NEW	
(ix) Set NEW > preu = PTR	
(X) Set NEW+ Next = NULL	4
(xi) Exit	6
	6
	*
	4
	5

4. Algo to Twent An Element After a particular location by data.
(i) If AVAIL == NULL, write overflow & Exit
(iv) Set AVAIL = AVAIL -> NEXT
(vi) Set PTR = PTR -> Next
(Viii) NEW -> prev = PTR -> Next
$(x) PTR \rightarrow Next \rightarrow PRW = NEW$ $(xi) PTR \rightarrow Next = NEW$ $(xi) Sxit$
5. Algo to Trust An Element Before a particular location by Data.
(i) If AVASIL == NULL, Write Overflow & Exit (ii) Set NEW= AVASIL
(iv) Set AUAIL= AVAIL -> Next (iv) Set NEW -> data = ITEM
(vi) Reseat Step (vii) While PTR -> data 1 = Data (of mode (vii) PTR = PTR -> Next
(VIII) Set PTR -> prew -> NEW (VX) Set NEW -> prew = PTR -> prew
(X) Set PTR -> PTCU = NEW (Xi) Set NEW -> PTR (Xii) Exit
(xii) Exit

```
6. Algo to delete first Node/Elemet
     IF START == NULL, Write Underflow, & Exit
(i)
 (ii) Set PTR = START
 (11) Set START = ST ART -> Next
 (W)
    SET START -> DOW = NULL
 (V) Free (PTR)
 (v) Exit
 7. Algo to delete Last Node of Linked
       List.
(i)
    If START == NULL, write underflow & Exit
(ii)
     SET PTR = STRRT
(ili)
    Repeat Step (iv) while PTR -> Next 1= NULL
(iv)
    Set PTR = PTR -> Next
(V) Set PTR -> Prev -> Next = NULL
 (vi) Free (PTR)
    Exit
 (iii)
    Algo to delete Element by Data.
 8.
(i)
    JE START == NULL, Write underflow & Exit
(ii)
     Set PTR - START
(iv) Repeat Step (iv) While PTR->data 1 = DATA
(iv)
    Set PTR = PTR > Next
(₩)
     Set PTR -> prev -> Next = PTR -> Next
(vi)
    Set PTR -> Next -> prew = PTR -> prew
(vii) Free (PTR)
 (viii) Erit
```