

(A)	Singly us Droubly Linked List:	
	10 720 2 PA 2 PA	
	7	
	head	
-	Advontages:	
	[00] (00) (00) (00) (00)	
1)	con be tooversed in both direction.  A given delete a note in O(1) time with given reference/pointer to it.  2 neert/Delete before a given rock.  I ansert/Delete from both end in O(1) time  by maintaining toil.	
2)	A given delete a node in O(1) time with given	
***	reference/pointer to it.	
3)	Insert/Delete before a given rock	
4)	I assert/Delete from both end in OCI) time	
	by maintaining toil.	
	U U	,
	i Carino	
	Disatuontages!-	
	. 1 1 1 7 7 1	,
	Extra space for Prev.	
2.		
	Then but active stall his the it's	
	1	
	: (n') ' Mey god 4	
· · · · · · · · · · · · · · · · · · ·		
	The second of the second of the second of	
	in the same in	
	,	



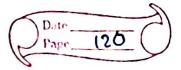
*	Insert at Beginning of & DLL
	The state of the s
,	neod
	IP:- 10 7 30
	data = 5
	Tro. arbo.
	OP1- [5] 7 10 7 20 7 30
	Till at head of processing the contract
	read.
	" al al action of the
	stouct Node Exist - and I itilized toward.
	int dato; it is the standing of the
•	Node * next;
	Node * Previ
	Node (int x) { datu = >c;
	Prev= WULLingsid
	next=NULU; }
	( 1 - 1 ) ( 21 )
	Node * insert Begin (Node * head, int data)
	Node * temp = new Node (data); temp -> next = head; if ( nead != NULL) { head -> prev = temp;}.
	temp -> next = head:
	if ( head != NULL) { head - prove = temp:}
	1 - 1 (m) - 10 (m) -
	return temp;
	7
	j
-	



( <del>x</del> )	Insert at End of OLLi-
(0)	2000 W
	IlP: - (10/7/20/7/30), data=40
	The state of the s
	0/P!- (10)=>20/=>40/
	1014-1014 CON (CO)
2	IlP:- NULL. datot=40
	Only case: where head gets change
	OP:- (40) head gets change
	Theod.
=>	
	Node *insert End (Node * khead, int data) [
	Node # temp = new Node (data);
	\\\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\
	2 return temp; ?
	Node * curr = head;
	while (curr -> next != NULL)
	t curr = curr -> next; 3
	con - ant - tours
	curr-> next = temp; temp-> prev = curr;
	76 111 2 2 2 20017
	return head;
	2
	3

		Date Page
( <del>1</del> )	Reverse a Doubly Linked List:	
)	IP: 10 220 7 30 7 40	
		W.
	OP: 40字30720710	· · ·
		11
2)	1 r:- 10	
	Line Control of the C	the Allie
	01:- [0]	
0)		
3)	Ilp:- NULL	
	OP:- NULL	· · · · · · · · · · · · · · · · · · ·
	We just how to swop the next &	-14 -3011
	previol all the element	and the
		261
3)	Nort 10 7 20 7 30 7 40 Next Next Next Next	ILL Many
		<u>) Hieror</u>
	Head : in the second of the second	
	ext Preu Preu Preu Preu	
NULL	-11) -120 7 30 -2 140-2 NU	LL
	Next Next Next	
	head	
	<u> </u>	
		V. max

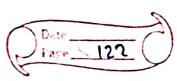
Time: Spore:	B(n)
spoce:	(1)0



=>	the state with the second second
	Node * reverse DLL (Node *head) {
	if Chead == NULL !! head => next == NULL)
	Freturn head; ?
	Node * Prev = NULL; cur Node * cur = head;
	while (curr!= NULL)
	[ Prev = curr -> Prev;
	curr -> Prev = Curr -> Next; (Swopping
	curr -> Next = Prev;
	Curr = Curr -> Prev; 3
	return prev -> Prev; &
	The said to the sa
	NULT 10 7 20 7 130 -> NULL
	After 13+ Devation:
-	After 1st Iteration:
n ti	i i i i i i i i i i i i i i i i i i i
	10 7 20 7 30 -> NULL PREV=NULL
	Turn ( )
	instant int
	After IInd Iteration:
	410 From 20 F 30)
	prev ann
	After III's Heration:
	TOPSNULL & [20/ NOUT [30] CURR=NULL)
	Totanuce 20/ 20/ Curr=NULL)



*	Delite Head of a Doubly Linked List:
)	了户: [10] 7[20] 7[30]
	OP:- 20 字30
2)	
	OP:- NULL
,	Time: 0(1)
3)	IP:- NULL
	OP:- NULL
• =>	
	Node * delHead (Node *head) { if Chead == NULL) {return Nuil; }
	if Chend = = NULL ) { xeturn NULL; }
	10000
	if Chead => Next == NUCU)
	delete head;
	return NULL; &
	else {
	Node *temp = head;
	head = head > next;
	$h \circ h \to h \circ $
	delete temp;
	delete temp;
	0
	4
•	



*	Delete dost Node of Doubly Linked dist:
1)	TP:- 10 全[20] 全[30] :
	0 P:- (10)=20 # (89)
2)	IP:- 110]
	Olp:- NULL
	Time: Ocn)
3)	IP:- NULL
	Olp:- NULL
<b>→</b>	
	Node * de Jast (Node *head) {
	if (head == NULL) { return NULL; }
	if ( head -> next == NULL) {
	delete headi
	who ise two inversity that without a se with he in
	Node * curr = head;
	while ( curr -> next ! = NULL)
	{ Curr = Curr -> Next;}
	Curr > prev > Next = NULL;
	delete currisers de
	seturn head;
	2
	Je we maintain a tail pointer then the loop is not sequired.
	tail pointer then
	The loop is two sequises.