	and the second of the second o
Lab 6.c	
#mclude a	
and the second server by the second s	Cstdlib.n>
struct 1	Node
<u>-</u>	
The same of the sa	_clata;
The state of the s	not now year
3;	node *head;
Stract	noue - rula /
void	
void	insert_last();
void	insert - handom();
uoi d	delete_beg();
void	delete - Last ();
Upid	display(); (11 ) and the state
int	item;
	, ( 3.2
uold	main () {
	int choice = 0;
	while (1) &
3	The state of the s
	printf C" * * MENU** \n" );
	CALL Clarace ALOUR COROLLY )
	print+ C" [1] Insert at Bog ining (1");
	printf ["[2] Insert at last (n");
Service of the	print f (" [u] Delete at beginning ("
	printf (" [3] In Belte at Regining \n");  printf (" [4] Delete at Begining \n");  printf (" [5] Delete at Last \n");
	printf C" [6] Delete at Kandom (1)
	printf ("[7] Display (n)
	printf ("[6] Delete at Random (n");  printf ("[7] Display (n");  printf ("[8] Exit\n");
2.12.54	

)	aff C !! Tax	(10")
	intf ("Enter your option: conf("Yd", & charce);	
SC SC	witch (choice) &	and the second s
	with Courses &	
	cay 1:	and the second s
	insert_beg();	a digital
	brunk;	
	case 2:	State to
	insent_layt();	Trus Establishmen
	break;	
	cose 3:	The state of the s
	insert trandom ():	
	bheak;	John Law Jus
	Can 4:	Walkel trains by
	dulete_beg();	1. Taylor trans the
	case 5:	4 9 81, 1 - 1,50
(		1941 - State June
	<u> delete nandom la</u> broalc	ast(); () algorithms do
C	are 6;	Sandi s lai
	dulete - man doni ();	
	bhack;	4.
	case 7:	100d) 10d
		9 (1) A.dei
	display(); break;	A - MA
	cax8! halp all	At Mark Artist
	exit(0);	Add to the first
	break;	1/17/1
	A Mark to to taken	
	clefault: printf C"In	Invalid Option 1 \n");
	7 .	priority (1)
Ž	30 reliable to a stable	
	- W. ()	

Scanned with CamScanner

```
uoid insert beg () &
      struct node *pth;

pth = (struct node *) malloc (size of (struct node));
       4 ( ptn = = NULL)
               phintf ("In Duartlow "In");
        else
              printf (" Enter value of node: > n");
             scanf (" Y.d", &item);
             periotal ptn -> duta = item;
                         ptn -> next = head;
head = ptn;
                        printf c" Node has been Successfully
                                   Inserted 1/ In ");
              inbort_late) {
       void
                     node * ptn, * temp;
             ptn = (struct node *) malloc(size of (struct node));
            y(ph = = NULL)
                        printf ("In Overflow ! ! ! \n");
              else 3
                     printf C" Enter value of Node::> (n ");
scanf C" Y.d", &item );
                       ptr -> data = item;
```

The continue of the continue o	ptn -> next = NULL;  head = pth;  print+L" Node successfully inserted	WH .	
The proximal party of the provided of the prov	U Chead == NULL;  head = pith;  printf(" Node successfully invented		
ptn => next = NULL;  head = pth;  printfl" Node successfully inserted	ptn -> next = NULL;  head = pth;  printf(" Node successfully inserted		- U Chiad == NVII)
ptn -> next = NULL; head = pth; printfel" Node successfully inserted                      temp = read; while (temp -> next = pth; pth -> next = pth; pth -> next = NULL; printfel" Node successfully Inferted! In  temp -> next = pth; pth -> next = NULL; printfel" Node successfully Inferted! In  int locat; pth                               pth = (Nitruct node *) melloc (size of (struct rode *) melloc (size of (stru	ptn -> next = NULL; head = pth; printfel" Node successfully inserted                      temp = read; while (temp -> next = pth; pth -> next = pth; pth -> next = NULL; printfel" Node successfully Inferted! In  temp -> next = pth; pth -> next = NULL; printfel" Node successfully Inferted! In  int locat; pth                               pth = (Nitruct node *) melloc (size of (struct rode *) melloc (size of (stru	动"——	3
Phintf(" Node successfully inserted ! In serted ! In s	Phintf(" Node successfully inserted ! In serted ! In s	***************************************	
else  temp = nead;  while (temp > next ! == NULL);  temp = temp > next : )  temp > next = ptn;  ptn > next = NULL;  pmintfl" Node Successfully Inherted! In  int locat;  thuck node * ptn * temp;  ptn = (Ntruct node *) malloc (size of (struct node)  y (ptn == NULL) ?  phintf ("In Overthown" (n");  }	else  temp = head;  while (temp -> next! == NULL);  temp -> next = ptn;  ptn -> next = NULL;  printf L'' Nodu Successfully Inserted! Ita  3  void ingent mandom() {  int locat;  ptn -= (Ntruct nodu *>) malloc (size of (streed rod)  if (ptn == NULL) {  phintf ("In Overthours!" \ n");  }	VI (IV	head = pth?
temp = head;  while (temp > next! == NULL);  temp = temp > next;  temp > next = ptn;  ptn > next = NULL;  printfl" Nodu Successfully Inherted! In  and ingular enandom() & temp;  ptn = (not node * ptn * temp;  ptn = (node * ptn * temp;  p	temp = head;  while (temp > next! == NULL);  temp = temp > next;  temp > next = pth;  pth > next = NULL;  printfl" Nodu Successfully Inherted! In  and Ingat mandom() &  int locat;  betweet node * pth * temp;  pth = (struct node *) malloc (size of (struct node) )  if (pth = = NULL) &  phintf ("In Overflow") *  phintf ("In Overflo	7	phintfl Noae giccestury
temp = read; while (temp -> rext! == NULL);  temp = temp -> rext;  temp -> rext = ptn; ptn -> rext = NULL; printf(" Node Successfully Inherted! The  printf(" Node Successfully Inherted! The  the locat; struct node *ptn * temp; ptn = (struct node *) malloc (size of (struct node));  ptn = NULL)?  ptnotf ("In Overflow!!" (n");  }	temp = head; while (temp -> Next ! == NULL);  temp = temp -> next;  temp -> next = ptn; ptn -> next = NULL; printf [" Node Succes fully Inherted ! The  printf [" Node Succes fully Inherted ! The  the locat; struct node *ptn * temp; ptn = (struct node *) malloc (size of (struct node ')		The state of the s
temp = head;  while (temp -> Next! == NULL);  temp = temp -> next;  temp -> next = ptn;  ptn -> next = NULL;  printf (" Nodu Success fully Inserted! In  tocat;  between the ptn + temp;  ptn = (not tocat + ptn + temp;  ptn = (not tocat + node + ptn + temp;  ptn = (node + node + node + ptn + temp;  ptn = (node + node + node + node + node + temp;  ptn = (node + node +	temp = head;  while (temp -> Next! == NULL);  temp = temp -> next;  temp -> next = ptn;  ptn -> next = NULL;  printf (" Nodu Success fully Inserted! In  tocat;  between the ptn + temp;  ptn = (not tocat + ptn + temp;  ptn = (not tocat + node + ptn + temp;  ptn = (node + node + node + ptn + temp;  ptn = (node + node + node + node + node + temp;  ptn = (node + node +	T	
while (temp > Next ! == NULL);  temp = temp > Next :  temp > Next = ptn;  ptn > Next = NULL;  pmintf(" Nodu Successfully Inherted! In  int locat;  btnuct nodu * ptn * temp;  ptn = (Null) ?  ptn = (Null) ?  phintf("In Overthown!"   n");  phintf("In Overthown!"   n");	while (temp > Next ! == NULL);  temp = temp > Next :  temp > Next = ptn;  ptn > Next = NULL;  pmintf(" Nodu Successfully Inherted! In  int locat;  btnuct nodu * ptn * temp;  ptn = (Null) ?  ptn = (Null) ?  phintf("In Overthown!"   n");  phintf("In Overthown!"   n");	The same of the sa	temp = head;
Imp = temp > rext;  temp >> rext = ptn; ptn >> rext = NVLL; pmintfl" Node Succesfully Inderted! In  int Ipcat; sthuct node *ptn *temp; ptn = (Ntruct node *) malloc (size of (struct node)  if (ptn = = NVLL)? phintf("In Workhow!"\n"); }	Imp = temp > rext;  temp >> rext = ptn; ptn >> rext = NVLL; pmintfl" Node Succesfully Inderted! In  int Ipcat; sthuct node *ptn *temp; ptn = (Ntruct node *) malloc (size of (struct node)  if (ptn = = NVLL)? phintf("In Workhow!"\n"); }	Notes	while (temp -> Nex+1 == NULL);
temp -> next = ptn;  ptn -> next = NVLL;  pmintfl" Nodu Success fully Inberted 1 [In  }  Void ingest frandom () {  Int locat;  betweet node * ptn * temp;  ptn = (1struct node *) malloc (size of (struct node i) (ptn = = NVLL) {  phintf ("In Overthown!" (n");  }	temp -> next = ptn;  ptn -> next = NVLL;  printfl" Nodu Successfully Inberted 1 [\n  }  void ingent trandom() \( \xi \)  int locat;  btnuct node * ptn * temp;  ptn = (pstruct node *) malloc (size of (struct node i) (ptn = = NVLL) \( \xi \)  phintf ("In Overthown!" \( \n^{\gamma} \);  }		3 Maria Administration J. House
temp -> next = pth;  pth -> next = NVLL;  pmintfl" Nodu Successfully Inherted 1 [In  2  2  3  Voi'd ingest transform () {  Int locat;  betweet node * pth * temp;  pth = (betweet node *) malloc (size of (struct node ) (pth = = NVLL) {  phintf ("In Overthows 11" (n"));  3  Phintf ("In Overthows 11" (n"));	temp -> next = pth;  pth -> next = NVLL;  pmintfl" Nodu Successfully Inherted 1 [In  2  2  3  Voi'd ingest transform () {  Int locat;  betweet node * pth * temp;  pth = (betweet node *) malloc (size of (struct node ) (pth = = NVLL) {  phintf ("In Overthows 11" (n"));  3  Phintf ("In Overthows 11" (n"));		그 사이 그렇게 그 아이에 나를 살아 가게 되는 것이 되었다. 그 그들은 그리고 살아보니까지 않는 것이 되었다. 그는 사이를 살아 있는 것이 없는 것이 없다는 것이 없다.
ptn -> next = NVLL;  printfl" Node Successfully Inserted 1 I/n  3  Void ingent mandom () &  Int locat;  Struct node * ptn * temp;  ptn = (struct node *) malloc (size of (struct node ) (ptn = = NVLL) &  Phintf ("In Overflow" (n");  3  Phintf ("In Overflow" (n");	ptn -> next = NVLL;  printfl" Node Successfully Inserted 1 I/n  3  Void ingent mandom () &  Int locat;  Struct node * ptn * temp;  ptn = (struct node *) malloc (size of (struct node ) (ptn = = NVLL) &  Phintf ("In Overflow" (n");  3  Phintf ("In Overflow" (n");		
printfl" Now successfully Inberted! In  2  Void ingent mandom () & Int locat; Int locat; Int locat; Int work * ptry * temp; Int = (Struct now *) malloc (size of (struct rob  y (ptr) = NULL) & Interpretation of the printf ("In Overthown!" \ n");  3  Printf ("In Overthown!" \ n");  3  Printf ("In Overthown!" \ n");	printfl" Now successfully Inberted! In  2  Void ingent mandom () & Int locat; Int locat; Int locat; Int work * ptry * temp; Int = (Struct now *) malloc (size of (struct rob  y (ptr) = NULL) & Interpretation of the printf ("In Overthown!" \ n");  3  Printf ("In Overthown!" \ n");  3  Printf ("In Overthown!" \ n");	N. T.	temp -> Next = pin )
void ingret mandom () \( \)  Int locat;  Struct node * ptn * temp;  ptn = (struct node *) malloc (size of (struct robe) \( \)	void ingret mandom () \( \)  Int locat;  Struct node * ptn * temp;  ptn = (struct node *) malloc (size of (struct robe) \( \)		pth > Next = NULL;
void ingrt_mandom() {  int locat;  struct node * ptn * temp;  ptn = (struct node *) mallox (size of (struct rod  i) (ptn = = NULL) {  phintf ("In Overthown!" (n");  }	void ingrt_mandom() {  int locat;  struct node * ptn * temp;  ptn = (struct node *) mallox (size of (struct rod  i) (ptn = = NULL) {  phintf ("In Overthown!" (n");  }		
Void ingest mandom () \( \)  Int locat;  Struct noch * ptr * temp;  ptr = (Struct noch *) malloc (size of (struct noch ) malloc (size of (struct noch ) malloc (size of (struct noch ) );  Print f ("In Overthown" \( \) \( \);	Void ingest mandom () \( \)  Int locat;  Struct noch * ptr * temp;  ptr = (Struct noch *) malloc (size of (struct noch ) malloc (size of (struct noch ) malloc (size of (struct noch ) );  Print f ("In Overthown" \( \) \( \);		2
int locat;  struct node * ptn * temp;  ptn = (struct node *) malloc (size of (struct node i) (ptn = = NULL) ?  phintf ("In Overthow!!" (n");  3	int locat;  struct node * ptn * temp;  ptn = (struct node *) malloc (size of (struct node i) (ptn = = NULL) ?  phintf ("In Overthow!!" (n");  3		3
Int locat;  struct node * ptn * temp;  ptn = (struct node *) malloc (size of (struct node  y (ptn = = NULL) {  phintf ("In Overthow!!" \ n");  }	Int locat;  struct node * ptn * temp;  ptn = (struct node *) malloc (size of (struct node  y (ptn = = NULL) {  phintf ("In Overthow!!" \ n");  }		
struct noch * ptn * temp;  ptn = (struct noch *) malloc (size of (struct noch  ŷ (ptn = = Null)?  phintf ("In Overflow!!"\n");  3	struct nock * ptn * temp;  ptn = (struct nock *) malloc (size of (struct noch  ŷ (ptn = = Null)?  phintf ("In Overthown!" (n");  3		WILL EMPOONING TE
ptn = (struct node *) mailor (size of (struct node ) (ptn = = NULL) & phintf ("In Overflow!!" \ n");	ptn = (struct node *) mailor (size of (struct node ) (ptn = = NULL) & phintf ("In Overflow!!" \ n");		Int locat;
Phintf ("In Overflow!!"\n");	Phintf ("In Overflow!!"\n");		Struct how * ptn * temp;
Phintf ("In Overflow!!"\n");	Phintf ("In Overflow!!"\n");		por = Chrynet node *) mailor (size of Cstruct rate
31 1019	31 1019		9 6 111-2 100 (1) 9
			Print ( In Worthows !! ( 1");
The state of the s	The state of the s		5
The state of the s	The state of the s		
			May a season by May a Mar Van and Salay
			10 mg/mg/mg/mg/mg/mg/mg/mg/mg/mg/mg/mg/mg/m

W. T. Links of the second of t
oua dulete last 1) &
Struct rock * pin, * phil ) 30 33 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4 (head == NULL)
7. 14. 14. 14. 14. 14. 14. 14. 14. 14. 14
printf ("In List in empty !!!! In");
N The state of the
else if Chead -> not == NULL) &
head = NULL; theo (head);
Printf C" Only Nock of List Deletion !!! In")
else
8
ptr = head;
while (ptn-> next1 = NULL) {
ptr1 = ptr
ptn = ptn -> next;
2110/A (110) A (1)
A Committee of the state of the
ptn 1 -> next = NULL;
free(ph);
prints ("In Deleted Node from Larth")
3 1651 - 1141 ( - 41)
3
3
<u> 1984년 김 아래, 1985년 1985</u> 전 1일

void delete_mandom () {
Struct node * ptn, * ptn1;
print L" Delete Location of Node to be deleted (n");
PHINTE (" Upiele Coccolor of 100 act to 12 con the V.
scant ("7.d", &locat);
ptn = head; for (int i = 0; i < locat; i++) ?
ton (int 1=0;1=1000)
olu I – voto :
ptal = pto ;
p+n = ph -> next; y(ph == NULL) \(\xi\)
oright ("(apport Delete 11 \n");
y(ptr == NULL) {  ptintf ("Cannot Delete   I \n");  3
3
ntn 1 -> next = ptn -> next ;
ptn 1 -> next = ptn -> next;  free (ptn);
print+ ("In Deleted Note: > 1.d", locat+1);
- 3
void display() {
struct node *ptx;
- blad'
y (ptn = = NULL) {  Phintf L"EMPTY !!! \n");
phintf L"EMPTY !!! (n");
3
else
Eq. (Section 1)
while(ptn! = NULL) {
pnint (" ", d In , ptn -> data)
pnintf(" ".d In", ptn -> data); ptn = ptn -> next;
3
3
3