Program 5

Write A Program to Implement Singly Linked List with following operations

- a) Create a linked list.
- b) Deletion of first element, specified element and last element in the list.
- c) Display the contents of the linked list.

```
Code:
_#include <stdio.h>
#include <stdlib.h>
struct Node {
  int data;
  struct Node* next;
};
struct Node* createNode(int data) {
  struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
  newNode->data = data;
  newNode->next = NULL;
  return newNode;
};
void insertatfirst(struct Node** head, int data){
  struct Node* newnode =createNode(data);
  newnode->next = *head;
  *head = newnode;
```

```
}
void deleteFirst(struct Node** head) {
  if (*head == NULL) {
    printf("The list is empty.\n");
    return;
  struct Node* temp = *head;
  *head = (*head)->next;
  free(temp);
void deleteElement(struct Node** head, int key) {
  if (*head == NULL) {
    printf("The list is empty.\n");
    return;
  }
  struct Node *temp = *head, *prev = NULL;
  if (temp != NULL && temp->data == key) {
    *head = temp->next;
    free(temp);
    return;
  while (temp != NULL && temp->data != key) {
```

```
prev = temp;
    temp = temp->next;
  }
  if (temp == NULL) {
    printf("Element %d not found.\n", key);
    return;
  prev->next = temp->next;
  free(temp);
}
void deleteLast(struct Node** head) {
  if (*head == NULL) {
    printf("The list is empty.\n");
    return;
  struct Node *temp = *head, *prev = NULL;
  if (temp->next == NULL) {
    *head = NULL;
    free(temp);
    return;
  }
  while (temp->next != NULL) {
    prev = temp;
```

```
temp = temp->next;
  }
  prev->next = NULL;
  free(temp);
}
void displayList(struct Node* head) {
  if (head == NULL) {
    printf("The list is empty.\n");
    return;
  }
  struct Node* temp = head;
  while (temp != NULL) {
    printf("%d -> ", temp->data);
    temp = temp->next;
  }
  printf("NULL\n");
}
int main() {
  struct Node* head = NULL;
  int choice, value;
  while (1) {
    printf("\nMenu:\n");
```

printf("1. Insert element at the end\n 2. Delete first element\n 3.Delete specified element\n 4.Delete last element\n 5.Display list\n 6.Exit\n");

```
printf("Enter your choice: ");
   scanf("%d", &choice);
switch (choice) {
      case 1:
        printf("Enter value to insert: ");
        scanf("%d", &value);
        insertatfirst(&head, value);
        break;
      case 2:
        deleteFirst(&head);
        break;
      case 3:
        printf("Enter value to delete: ");
        scanf("%d", &value);
        deleteElement(&head, value);
        break;
      case 4:
        deleteLast(&head);
        break;
      case 5:
        displayList(head);
        break;
```

```
case 6:

exit(0);

default:

printf("Invalid choice.\n");

}

return 0;
```

1. Insert element at the end Delete first element Delete specified element 4.Delete last element 5.Display list 6.Exit Enter your choice: 2 Menu: 1. Insert element at the end 2. Delete first element 3.Delete specified element 4.Delete last element 5.Display list 6.Exit Enter your choice: 5 38 -> 23 -> 14 -> NULL Menu: 1. Insert element at the end 2. Delete first element 3.Delete specified element 4.Delete last element 5.Display list Exit Enter your choice: 4

```
1. Insert element at the end
2. Delete first element
3.Delete specified element
4.Delete last element
 5.Display list
 6.Exit
Enter your choice: 1
Enter value to insert: 38
1. Insert element at the end
 2. Delete first element
3.Delete specified element 4.Delete last element
 5.Display list
6.Exit
Enter your choice: 1
Enter value to insert: 45
1. Insert element at the end
 2. Delete first element
 3.Delete specified element
4.Delete last element
5.Display list
 6.Exit
Enter your choice: 5
45 -> 38 -> 23 -> 14 -> NULL
```

```
1. Insert element at the end
 2. Delete first element
 3.Delete specified element
4.Delete last element
 5.Display list
 6.Exit
Enter your choice: 5
38 -> 23 -> NULL
Menu:
1. Insert element at the end
 2. Delete first element
 3.Delete specified element
4.Delete last element
 5.Display list
 6.Exit
Enter your choice: 3
Enter value to delete: 23
1. Insert element at the end
 2. Delete first element
 3.Delete specified element
 4.Delete last element
 5.Display list
 6.Exit
Enter your choice: 5
38 -> NULL
```

. The side Has Li.	store 67
WAP to implement singly linked live with the fel	and the same of th
Operations.	Since mode + keng - + head;
a) Create a linked lise	KILLOW: FIN CAMPLE
a) Create a linked list brush & love downer by Polishon of fact element specified element & love downer	
in the late	free temp);
c) Propay the contract of the lacked list	return;
c) Waging the course of the	3
	while (kung & next + next != NULL) {
#include Cathie W	time = temper nerti
# include <abblishing< td=""><td>1 Overall 1</td></abblishing<>	1 Overall 1
Smit Node 9	free (sump = next);
int data;	town > next . NULL:
Smuck hode + next;	11
1;	word delete At Position (struct Node** head, int position) ?
Street Node * Create Node (but date) { Street Node * Mr. Node = (street Node *) maker (size of) greet Node * Mr. Node = (street Node *) maker (size of)	if (*head == NULL) {
giruit Node " new Node = Christian Greekland	print "List is empty")
	\$ control of
phis Node -> data = data;	Struct Node + semp = + head;
NEW Node -> vest = NULL;	if (position == 0)9
Tehurn neu Nodes 1996 10 100 1	duct Francist (head)
void delete from first (street Node** head) {	(ckunt)
	Windshide styres abulkans
print ("List is empty "a");	for line i=D; lemp!= Nul to icposition -1; i++) }
folunt.	Hump kunp & next;
	The state of the s
Sour Note + sup = + head;	if Camp == NULL 11 kmp + next == NULL)?
* head = jourp = nest;	print ("Position out of Tange")
fere (xing);	return;
1	
Vaid delek from Red Shace Node ** head &	smet Node + next + tunp = next + next;
is (*head == Nove)	free (ump -) next is a read to the stand of the
pivits ("List is empty");	kmp = next = next; [mes a bear transmit states
plant;	13 Stopp & Variet Fried
The state of the s	

3	store 67/
word issure Affire (struce Node ** head; int data?	kmp -> next- newword;
void insert Hist (struck Node + Wass (date);	The state of the s
Smit Nodet New Node - Creater	void point (struct Node + head) ?
New Node 7 466 = + here	Street Node * kup = head ;
+ hend = new Node;	whilelieung = NOWF
Void invited Position (Start Node ** Wad, int position;	of Citation day orders)
Void invol H Position (Struck Node + 2 record	kurp=kurp - next;
Street Node + new Node = (rooks Node (data);	3 and the second second (and a second
Smot Node + newhole = (make 1000)	print ("NOL");
il fortion == 0) }	Talas Caul Protei shirt
in postion = (Chead, and);	
Rium:	int main() 9
	shuck Node* head . Null;
Smuth Node + kup = + head; For Cintis O; kup = NML + 1 < position - 1, 1+1)9	lak choice, data;
For lines O; level = NULLER 12 porter	whilely 2 was banks many statute
kurg = lunp = MUL;	printf ("Linked list operations:");
	privit ("I herrion at first position in 2- Invertion at
if (tup:=Nut) & marin and a marin	Specified possion In 3- Lincolon at End to 4- Popular
news Vesition Stat of Takes	I'm 5. Deretion alend In"
free (new Node);	pl("Ener your choice")
	scant ("d", f cho; e);
newNode -> next = kup -> next;	quark (choice) 9
kup - nex 6 - new frode;	
3 0 10 1 × × 1 1 1 1 do 608	case 1: printf "Entr doment to be inscited it in"
Void incit Ht End (struct Node * + head, just darin)?	sconf ("+d", Ldata);
greek Node + New Node > arate Node (data);	increat first (I head , data);
if (+ head == NULL) {	bilat, and
* Weed - New Wede ?;	case 2: of "Enter element to the inscribed: \");
	st ("Ita", Lanla);
CALL TO THE WAY A VALUE OF THE PARTY OF THE	insert Afgesition (I head, position, data);
Street Nede & king = *head.	Brenki
while (ump-> min = NULL) ?	
g lamp = lamp = next;	

-	store 67
(4)	Linked Line Operations
case 3: printell "Enter element brisses at the ene.")	1- Insertion at first Position
34 ("-1-d", &dala);	2. Insertion at Specified Position
issert At End (Aled, date);	S. Insertion at find
I Word At Lind Caneed, Onto	y. Display
briak;	5. Peletin at Enr position
Core 4: privet()	6. Deletion at specified position.
biak;	7. Deletion at end.
1 1. Jateus.")	Ever your Choice: 1
Case 5: Point ("tury demont to be deleyd:");	Ever element to be inscribed 12
delete Hine (Theod, data);	Ever your choice: 2
broks	Easter position and alement to be inserted: 2 14
	total parties and member to be inserted: 2 14
Case b: printf "Enter element to be deleted:");	Cubr your choice: 3
- CA' 1 1" Edgle):	
delet typosian (thread, position, data).	Enter Dement to be inserted at the end: 9
brakj	Enter your Choice: 5
As a post of record to the first to the second to the seco	
rear 7: ming ("Ener element to be delekt ");	Enter your choice: 4
gout ("Vid", Konta),	12 - 14 - 9 - NUL
delete At find [Allend stare);	
Mali	fuer your choice: 6
Grat)	Cuter position to delete 2
the state of the s	
default : printf ("Insolid choise"),	Enter your choice: 7
3	N.
3 Now 19 years	Marke My
Va dill	Low White
and the second	
the second services with the second services and the second services and the second services are second services as the second services are second services a	