Program 4

Write A Program to Implement Singly Linked List with following operations

- a) Create a linked list.
- b) Insertion of a node at first position, at any position and at end of list.
- c) Display the contents of the linked list.

```
Code:
#include <stdio.h>
#include <stdlib.h>
struct Node {
       int data;
       struct Node* next;
};
void createList(struct Node** head);
void insertAtBeginning(struct Node** head, int data);
void insertAtPosition(struct Node** head, int data, int position);
void insertAtEnd(struct Node** head, int data);
void displayList(struct Node* head);
int main() {
       struct Node* head = NULL;
       int choice, data, position;
       while (1) {
```

```
printf("\nMenu:\n");
printf("1. Create a linked list\n");
printf("2. Insert at the beginning\n");
printf("3. Insert at a specific position\n");
printf("4. Insert at the end\n");
printf("5. Display the list\n");
printf("6. Exit\n");
printf("Enter your choice: ");
scanf("%d", &choice);
switch (choice) {
  case 1:
     createList(&head);
     break;
  case 2:
     printf("Enter data to insert at the beginning: ");
     scanf("%d", &data);
     insertAtBeginning(&head, data);
     break;
  case 3:
     printf("Enter data to insert: ");
     scanf("%d", &data);
     printf("Enter position to insert (starting from 1): ");
```

```
scanf("%d", &position);
         insertAtPosition(&head, data, position);
          break;
       case 4:
         printf("Enter data to insert at the end: ");
          scanf("%d", &data);
         insertAtEnd(&head, data);
         break;
       case 5:
         displayList(head);
          break;
       case 6:
         printf("Exiting the program.\n");
         exit(0);
       default:
         printf("Invalid choice. Please try again.\n");
       }
       return 0;
}
void createList(struct Node** head) {
```

```
int data, choice;
       do {
    printf("Enter data to insert: ");
    scanf("%d", &data);
     insertAtEnd(head, data);
    printf("Do you want to add another node? (1 for Yes, 0 for No): ");
    scanf("%d", &choice);
       \} while (choice != 0);
}
void insertAtBeginning(struct Node** head, int data) {
       struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
       if (newNode == NULL) {
     printf("Memory allocation failed.\n");
    return;
       }
  newNode->data = data;
  newNode->next = *head;
       *head = newNode;
  printf("Node inserted at the beginning.\n");
}
void insertAtPosition(struct Node** head, int data, int position) {
```

```
if (position < 1) {
  printf("Invalid position.\n");
  return;
     }
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    if (newNode == NULL) {
  printf("Memory allocation failed.\n");
  return;
     }
newNode->data = data;
    if (position == 1) {
  newNode->next = *head;
  *head = newNode;
  printf("Node inserted at position %d.\n", position);
  return;
     }
    struct Node* temp = *head;
    for (int i = 1; i < position - 1 && temp != NULL; <math>i++) {
  temp = temp->next;
     }
```

```
if (temp == NULL) {
    printf("Position out of bounds.\n");
    free(newNode);
    return;
       }
  newNode->next = temp->next;
  temp->next = newNode;
  printf("Node inserted at position %d.\n", position);
}
void insertAtEnd(struct Node** head, int data) {
      struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
      if (newNode == NULL) {
    printf("Memory allocation failed.\n");
    return;
  newNode->data = data;
  newNode->next = NULL;
      if (*head == NULL) {
    *head = newNode;
      } else {
```

```
struct Node* temp = *head;
    while (temp->next != NULL) {
       temp = temp->next;
    temp->next = newNode;
       }
       printf("Node inserted at the end.\n");
}
void displayList(struct Node* head) {
       if (head == NULL) {
    printf("The list is empty.\n");
    return;
  printf("Linked list contents: ");
       struct Node* temp = head;
       while (temp != NULL) {
    printf("%d -> ", temp->data);
    temp = temp->next;
       }
  printf("NULL\n");
}
```

1. Create a linked list 2. Insert at the beginning 3. Insert at a specific position 4. Insert at the end 5. Display the list 6. Exit Enter your choice: 2 Enter data to insert at the beginning: 12 Node inserted at the beginning. Menu: 1. Create a linked list 2. Insert at the beginning 3. Insert at a specific position 4. Insert at the end 5. Display the list Exit Enter your choice: 2 Enter data to insert at the beginning: 23 Node inserted at the beginning. 1. Create a linked list 2. Insert at the beginning 3. Insert at a specific position 4. Insert at the end 5. Display the list 6. Exit Enter your choice: 3 Enter data to insert: 2 Enter position to insert (starting from 1): 2 Node inserted at position 2.

Menu:

- 1. Create a linked list
- 2. Insert at the beginning
- 3. Insert at a specific position
- 4. Insert at the end
- 5. Display the list
- 6. Exit

Enter your choice: 5

Linked list contents: 23 -> 2 -> 12 -> NULL

Menu:

- 1. Create a linked list
- 2. Insert at the beginning
- 3. Insert at a specific position
- 4. Insert at the end
- 5. Display the list
- 6. Exit

Enter your choice: 6
Exiting the program.

THE REAL PROPERTY AND PERSONS ASSESSED.	The state of the s
optenent a singly lined list with operations insect	for line is 1; 1 < pox 1 of temp 1 = NOA ; is >) }
recogning , end , display contents	preliate: 1: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
The same of the sa	temp - semp -> nexts
THE RESERVE OF THE PARTY OF THE	
Hirdude Shidio'h	if (Hup =: NULL) {
#include <adhib hi<="" td=""><td>if (thup :: Noul) } print ("Position ask of range \n");</td></adhib>	if (thup :: Noul) } print ("Position ask of range \n");
	The same of the same
Greek Node (else E
int date; " And a sun to state	mentioned a next a lease a react
Struct Node * nech;	true > neet = new Nede;
2.	truct . Inter-
South Node + createNode (ine dasa)?	75
Street Node " reservade (un assay) Street Node " reservade (comit Node *) Walloc (size of street of the	void insert at end (street Modet head?, int date)?
	void insert at find issortion whole: create Node (data)
new Node - date - date;	Should had a help been
newNode - nest = nulls	
yeturn newNede;	+ head = newNoole;
to the second se	Yekurw.;
The American Conference of the	
void insert at beginning (struct Node ** head, int data) f	Clark in his ar and displayment to make his
struce Node + newNode - cresteNede (data);	Struck Node + temp = + head
newhode -> next = + head;	while (tous + Mext = NOLL)
* head > new Node; and a wall and	temp -temp + nexti
	1
Phillips on the goods	true - new = newplade;
upid incert, at position (struct Node * A head, int data, int gos) f	The second secon
11(905=1)9	/ 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
neurole - nece - threat	void display (struct Node + nead) f
+ bead = newNoole;	
rount;	print ("Empty List \n");
3	The state of the s
Smit Nede + Jewy : + head;	Yellumi

stre 67	24
Shuz Node + Jung = Med;	
while (temp != None) {	The second second second second
while (temp !: hour) f	Enter your choice : 1
lamp - lemp + nect;	Enter value to insert at the beginning: 2
The state of the s	(Alab = April 10 and 1
printf ("NULL In");	Enkr your choice 2
3	Enter value to insert at the find: 3
int main() for any and a second	Carama Ca
Struct Node * head · Nuez	Eukr your choice: 5
WHALL INDE THESE TOURS	face who is impress to the togething
find at her a law to the	lived liss: 2 73 - NOLL
inserte at beginning (Mead, 6); display (Mend);	
auspau (naid),	Enter your choice th
	Enter your choice is
intert. ac. pos (A) read , 15,3);	Exiting congrams
display Chend);	
	of the state of th
injust-at- end (fread, 30);	
display (head);	
	(OZ BANK) has a grad
Ythen 0;	Charles Market
3	
M AL	
Bernete W.	
Name 28/10/	The state of the s
	Many the same
	and the second s
Output	
Hena:	
1 lovert at hegicaling	A PARTY OF THE PAR
2. Ingert at end	
3. Display the list	