```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<process.h>


struct Node
{
	struct Node *prev;
	int ele;
	struct Node *next;
};

struct Node *first, *last;

void insert_node()
{
	struct Node *nn, *temp, *temp2;
	int ch, sele, flag;
	nn = (struct Node *) malloc(sizeof(struct Node) );
	printf("Enter element for New Node : ");
	scanf("%d", &nn->ele);

	if(first == NULL)
	{
		nn->next = nn;
		nn->prev = nn;
		first = nn;
		last = nn;
		printf("List Created..\n");
	}
	else
	{
	  // means list is already created.
	  printf("Where you want to place this New Node : ");
	  printf("1 - At First Position\n");
	  printf("2 - At Last Position\n");
	  printf("3 - At Specific Position\n");
	  printf("Provide your choice : ");
	  scanf("%d", &ch);

	  switch(ch)
	  {
```

```
case 1: //inserting nn at first position
        nn->prev = last;
        nn->next = first;
        first->prev = nn;
        last->next = nn;
        first = nn;
        printf("New Node inserted at First Position.\n");
        break;
case 2: // inserting nn at last position
        nn->prev = last;
        nn->next = first;
        last->next = nn;
        first->prev = nn;
        last = nn;
        printf("New Node inserted at Last Position.\n");
        break;
case 3: // inserting nn at specific position
        printf("Enter element of that, after which you want to place NN : ");
        scanf("%d", &sele);
        flag = 1;
        temp = first;
        do
        {
                if(temp->ele == sele)
                {
                        flag = 2;
                        break;
                }
                temp = temp->next;
        }while(temp!=first);

        if(flag==1)
        {
                printf("No such node found.\n");
        }
        else if(flag==2 && temp==last)
        {
          // means selected node is last node.
          nn->prev = last;
          nn->next = first;
          last->next = nn;
          first->prev = nn;
          last = nn;
```

```
                printf("New Node inserted after Specified Node.\n");
              }
              else
              {
                //means selected node is other than last node.
                temp2 = temp->next;
                nn->prev = temp;
                nn->next = temp2;
                temp->next = nn;
                temp2->prev = nn;
                printf("New Node inserted after Specified Node.\n");
              }
              break;
          } //end of switch-case
        } //end of outer else
} //end of insert_node()

void remove_node()
{
        int ch, sele, flag;
        struct Node *temp, *temp2, *temp3;

        if(first == NULL)
        {
                printf("List Underflow. Yet Not Created.\n");
        }
        else
        {
          if(first == last)
          {
            temp = first;
            first = NULL;
            last = NULL;
            free(temp);
            printf("There was only one node. It is now removed.\n");
          }
          else
          {
            //means there are multiple nodes.
            printf("Which node you want to Remove ?\n");
            printf("1 - Remove First Node\n");
            printf("2 - Remove Last Node\n");
            printf("3 - Remove Specific Node\n");
```

```
printf("Provide your choice : ");
scanf("%d", &ch);
switch(ch)
{
	case 1: //remove first node
		temp = first;
		first = first->next;
		first->prev = last;
		last->next = first;
		free(temp);
		printf("First node is Removed.\n");
		break;
	case 2: // remove last node
		temp = last;
		last = last->prev;
		last->next = first;
		first->prev = last;
		free(temp);
		printf("Last node is Removed.\n");
		break;
	case 3: // remove specific node
		printf("Enter element of that node, which you want to remove : ");
		scanf("%d", &sele);
		flag = 1;
		temp = first;
		do
		{
			if(temp->ele == sele)
			{
				flag = 2;
				break;
			}
			temp = temp->next;
		}while(temp!=first);

		if(flag==1)
		{
			printf("No such node found.\n");
		}
		else if(flag==2 && temp==first)
		{
		  //means selected node is first node.
		  first = first->next;
```

```
                        first->prev = last;
                        last->next = first;
                        free(temp);
                        printf("Specified Node is Removed.\n");
                      }
                      else if(flag==2 && temp==last)
                      {
                        //means selected node is last node
                        last = last->prev;
                        last->next = first;
                        first->prev = last;
                        free(temp);
                        printf("Specified Node is Removed.\n");
                      }
                      else
                      {
                        //means selected node is in-between
                        temp2 = temp->prev;
                        temp3 = temp->next;
                        temp2->next = temp3;
                        temp3->prev = temp2;
                        free(temp);
                        printf("Specified Node is Removed.\n");
                      }
                      break;
                } //end of switch-case
              }
            } //end of outer else
} //end of remove_node()

void display_list()
{
        struct Node *temp;
        if(first == NULL)
        {
          printf("List is yet not created. Nothing to display.\n");
        }
        else
        {
          printf("List Contains\n");
          temp = first;
          do
          {
```

```
                printf("%d\t", temp->ele);
                temp = temp->next;
          }while(temp!=first);
        }
} //end of display_list()

void main()
{
        int ch;
        clrscr();
        first = NULL;
        last = NULL;

        while(1)
        {
                getch();
                clrscr();
                printf("Select operation\n");
                printf("1 - Insert Node\n");
                printf("2 - Remove Node\n");
                printf("3 - Display List\n");
                printf("4 - EXIT\n");
                printf("Provide your choice : ");
                scanf("%d", &ch);

                switch(ch)
                {
                        case 1: insert_node();
                                break;
                        case 2: remove_node();
                                break;
                        case 3: display_list();
                                break;
                        case 4: exit(0);
                } //end of switch-case
        } //end of while

        getch();

}
```