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


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

struct Node *first;

void insert_node()
{
	struct Node *nn, *temp;
	int ch, sele;

	nn = (struct Node *) malloc(sizeof(struct Node) );
	printf("Enter element for New Node : ");
	scanf("%d", &nn->ele);

	if(first == NULL)
	{
		nn->next = NULL;
		first = nn;
		printf("List Created..\n");
	}
	else
	{
	  // means list is already created. Show the choices.
	  printf("Where you want to place this New Node ?\n");
	  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->next = first;
			first = nn;
```

```
                printf("New Node is placed at First Position.\n");
                break;
        case 2: // inserting nn at last position
                temp = first;
                while(temp->next != NULL)
                {
                        temp = temp->next;
                } //at the end "temp" points to last node

                nn->next = NULL;
                temp->next = nn;
                printf("New Node is Placed at Last Position\n");
                break;
        case 3: // inserting nn at specific position
                printf("Enter element of that node, after which you wnant to insert NN : ");
                scanf("%d", &sele);

                temp = first;
                while(temp->ele != sele && temp!=NULL)
                {
                        temp = temp->next;
                }

                if(temp==NULL)
                {
                        printf("No such node found.\n");
                }
                else
                {
                        // means selected node found.
                        // "temp" points to selected node.

                        nn->next = temp->next;
                        temp->next = nn;
                        printf("New Node is placed after Specified Node.\n");
                }
                break; //end of case-3
          } //end of switch-case
        } //end of outer else
} //end of insert_node()

void remove_node()
{
```

```
struct Node *temp, *temp2;
int ch, sele;

if(first == NULL)
{
        printf("List Underflow.\n");
}
else
{
        // means list is already created.
        if(first->next == NULL)
        {
          // means there is only one node.
          temp = first;
          first = NULL;
          free(temp);
          printf("There was only one Node. It is now Removed.\n");
        }
        else
        {
          // means there are more than 1 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: // removing first node
                  temp = first;
                  first = first->next;
                  free(temp);
                  printf("First Node is Removed.\n");
                  break;
            case 2: // removing last node
                  temp = first;
                  while(temp->next != NULL)
                  {
                        temp = temp->next;
                  } // at the end "temp" points to last node
```

```
        temp2 = first;
        while(temp2->next != temp)
        {
                temp2 = temp2->next;
        } //at the end "temp2" points to second-last node

        temp2->next = NULL;
        free(temp);
        printf("Last node is Removed.\n");
        break;
case 3: // removing specific node
        printf("Enter element of that node, which you want to Remove : ");
        scanf("%d", &sele);

        temp = first;
        while(temp->ele != sele && temp != NULL)
        {
                temp = temp->next;
        }

        if(temp==NULL)
        {
                printf("No such node found.\n");
        }
        else
        {
                // means specific node found.
                // "temp" points to specific node.

                if(temp==first)
                {
                        //means selected node is first node.
                        first = first->next;
                        free(temp);
                        printf("Specified node is Removed.\n");
                }
                else
                {
                  // selected node is other than first node
                  temp2 = first;
                  while(temp2->next != temp)
                  {
                        temp2 = temp2->next;
```

```
                                        }
                                        // at the end "temp2" points to selected node's previous node
                                        temp2->next = temp->next;
                                        free(temp);
                                        printf("Specified node is Removed.\n");
                                    }
                                }
                                break; //end of case-3
                            } //end of switch-case
                        } // end of inner else
                } //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;
                while(temp!=NULL)
                {
                        printf("%d  ", temp->ele);
                        temp = temp->next;
                }
        } //end of outer else
} //end of display_list()

void main()
{
        int ch;
        clrscr();
        first = NULL;
        while(1)
        {
                getch();
                clrscr();
                printf("Select Operation\n");
```

```
		printf("1 - INSERT NEW 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 loop

	getch();
}
```