DATE : 20.12.2022

**USN: 1NT21IS117**

1. Design and implement a stack (linked list implementation) and demonstrate its working with necessary inputs. Display the appropriate messages in case of exceptions.

ALGORITHM:-

Aim: To implement stack using linked list.

Algorithm:

1. For push operation:

• Create a node

• Check if stack is empty

• If empty, set pointer node to null

• Else make node point to head

• Create new node as head

2. For pop operation:

• Check if stack is empty

• If empty, display ‘empty stack’

• Else set head to temporary node

• Print the data of head

• Make head point to next node

3. For display operation:

• Create temporary head

• Initialize it with head pointer

• Check if stack is empty

• If empty, display ’empty stack’

• Else transfer the temporary node

PROGRAM:-

#include<stdio.h>

#include<conio.h>

struct Node

{

int data;

struct Node \*next;

}

\*top = NULL;

void push(int);

void pop();

void display();

void main()

{

int choice, value;

clrscr();

printf("\n:: Stack using Linked List ::\n");

while(1){

printf("\n\*\*\*\*\*\* MENU \*\*\*\*\*\*\n");

printf("1. Push\n2. Pop\n3. Display\n4. Exit\n");

printf("Enter your choice: ");

scanf("%d",&choice);

switch(choice){

case 1: printf("Enter the value to be insert: ");

scanf("%d", &value);

push(value);

break;

case 2: pop(); break;

case 3: display(); break;

case 4: exit(0);

default: printf("\nWrong selection!!! Please try again!!!\n");

}

}

}

void push(int value)

{

struct Node \*newNode;

newNode = (struct Node\*)malloc(sizeof(struct Node));

newNode->data = value;

if(top == NULL)

newNode->next = NULL;

else

newNode->next = top;

top = newNode;

printf("\nInsertion is Success!!!\n");

}

void pop()

{

if(top == NULL)

printf("\nStack is Empty!!!\n");

else{

struct Node \*temp = top;

printf("\nDeleted element: %d", temp->data);

top = temp->next;

free(temp);

}

}

void display()

{

if(top == NULL)

printf("\nStack is Empty!!!\n");

else{

struct Node \*temp = top;

while(temp->next != NULL){

printf("%d--->",temp->data);

temp = temp -> next;

}

printf("%d--->NULL",temp->data);

}

}

OUTPUT:-



