Experiment No: 6 Date: 24 Nov 2021

Aim: To implement a stack using a singly linked list

Algorithm:

1. START
2. Declare a structure containing a data part as well as an address part
3. Present a menu of operations push,pop,display to the users by switch-case
4. If the operation is push()

4.1. Check whether array is empty or not

4.1.1. If empty create a node in newnode and assign TOP=newnode

4.1.1. Store a value in newnode->data

4.1.2. Else, create a temporary node and assign struct node \*temp=TOP

4.1.2.1. While tem->next!=NULL, traverse the linked stack

4.1.2.1.1. Attach a newnode with a value at the end position

4.2. Go back to step 3

1. If operation is pop()

5.1. Check whether list is empty or top is NULL

5.1.1 If yes, print ‘ stack underflow’, else free the last element by iterating list.

5.2. Display the linked stack elements

5.3. Go back to step 3

1. If operation is display, traverse the list items one by one and print the values
2. If operation is for exit, quit the menu and return
3. STOP

Program:

#include<stdio.h>

#include<stdlib.h>

void push();

void pop();

void display();

struct stacknode

{

int data;

struct stacknode \*next;

}\*top=NULL;

void main()

{

int opt;

do

{

printf("\n SELECT AN OPTION FROM THE BELOW MENU\n");

printf("\n 1. PUISH OPERATION\n");

printf("\n 2. POP OPERATION\n");

printf("\n 3. VIEW LINKED STACK\n");

printf("\n 4. EXIT\n");

scanf("%d",&opt);

switch(opt)

{

case 1: push();

break;

case 2: printf("\n Poping Out the last item :\n");

pop();

break;

case 3: printf("\n The stack consists of items \n");

display();

break;

case 4: exit(0);

default: printf("\n Invalid Option\n");

}

}

while(opt!=4);

}

void push()

{

struct stacknode \*newnode;

newnode=(struct stacknode\*)malloc(sizeof(struct stacknode));

printf("\n enter a value :\n");

scanf("%d",&newnode->data);

if(top==NULL)

{

newnode->next=NULL;

top=newnode;

}

else

{

newnode->next=top;

top=newnode;

}

printf("\n Now, the items in the stack are\n");

display();

}

void pop()

{

if(top==NULL)

printf("\n Stack Underflow, Insert atleast one item\n");

else

{

struct stacknode \*temp=top;

top=temp->next;

printf("\n the item %d has been poped out...\n",temp->data);

free(temp);

printf("\n Now , the items in the stack are\n");

display();

}

}

void display()

{

struct stacknode \*temp=top;

if(top==NULL)

printf("\n STACK IS EMPTY!\n");

else

{

while(temp!=NULL)

{

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

temp=temp->next;

}

}

}

OUTPUT: