LAB 7 DSA

```
/*Write a menu driven program to perform operation stack adt
1.Push
2.Pop
3.Display
4.Sort
5.Add a element to sorted stack
6.Find out the max element in the stack */
#include<stdio.h>

#include<stdlib.h>

#define MAX 5

int top=-1, stack[MAX];
void push();
void pop();
void display();


int main()
{
int ch;
while(1)
{
printf("\n*** Stack Menu ***");
printf("\n\n1.Push\n2.Pop\n3.Display/n4.Exit");
printf("\n\nEnter your choice(1-4):");
scanf("%d",&ch);
switch(ch)
{
case 1: push();
break;
case 2: pop();
break;
case 3: display();
break;
case 4: exit(0);
default: printf("\nWrong Choice!!");
}
}
}

void push()
{
int val;
if(top==MAX-1)
{
printf("\nStack is full!!");
}
else
{
printf("\nEnter element to push:");
scanf("%d",&val);
top=top+1;
stack[top]=val;
}
}

void pop()
{
if(top==-1)
{
printf("\nStack is empty!!");
}
else
{
printf("\nDeleted element is %d",stack[top]);
top=top-1;
}
}

void display()
{
int i;
if(top==-1)
{
printf("\nStack is empty!!");
}
else
{
printf("\nStack is...\n");
for(i=top;i>=0;--i)
printf("%d\n",stack[i]);
}
}
```

OUTPUT

```
*** Stack Menu ***

1.Push
2.Pop
3.Display/n4.Exit

Enter your choice(1-4):1

Enter element to push:4

*** Stack Menu ***

1.Push
2.Pop
3.Display/n4.Exit

Enter your choice(1-4):1

Enter element to push:5

*** Stack Menu ***

1.Push
2.Pop
3.Display/n4.Exit
```

```
Enter your choice(1-4):6

Wrong Choice!!
```

Write a menu driven program to implement STACK ADT [PUSH, POP &amp; DISPLAY] using single linked list.

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

struct Node {
   int data;
   struct Node *next;
}*start=NULL;



int pop() {
   if (start == NULL) {
      printf("\nEMPTY STACK");
   }
   else{
      struct Node *temp = start;
      int temp_data = start->data;
      start = start->next;
      free(temp);
      return temp_data;
   }
}

void push(int value) {
   struct Node *n = (struct Node *)malloc(sizeof(struct Node));
   n->data = value;
   if (start == NULL) {
      n->next = NULL;
   } else {
      n->next = start;
   }
   start = n;
   printf("Node is Inserted\n\n");
}

void display() {

   if (start == NULL) {
      printf("Underflow!!\n");
   } else {
      printf("The stack data is: \n");
      struct Node *temp = start;
      while (temp->next != NULL) {
         printf("%d\t", temp->data);
         temp = temp->next;
      }
      printf("%d\t\n\n", temp->data);
   }
}
```

```c
int main() {
   int choice, value;
   printf("\n Stack using Linked List\n");
   while (1) {
      printf("1. Push\n2. Pop\n3. Display\n4. Exit\n");
      printf("\nEnter your choice : ");
      scanf("%d", &choice);
      switch (choice) {
      case 1:
         printf("\nEnter the value to insert: ");
         scanf("%d", &value);
         push(value);
         break;
      case 2:
         printf("Popped element is :%d\n", pop());
         break;
      case 3:
         display();
         break;
      case 4:
         exit(0);
         break;
      default:
         printf("\n Enter correct Choice!!\n");
      }
   }
}
```

OUTPUT

```
Stack using Linked List
1. Push
2. Pop
3. Display
4. Exit

Enter your choice : 1

Enter the value to insert: 5
Node is Inserted

1. Push
2. Pop
3. Display
4. Exit

Enter your choice : 1

Enter the value to insert: 7
```

```
Node is Inserted

1. Push
2. Pop
3. Display
4. Exit

Enter your choice : 1

Enter the value to insert: 7
Node is Inserted

1. Push
2. Pop
3. Display
4. Exit

Enter your choice : 3
The stack data is:
7       7       5

1. Push
2. Pop
3. Display
4. Exit

Enter your choice : 4
```

Rishikesh
2105734
CSE 32