**Write a C++ Program to implement Stack using linked list?**

* A stack is an abstract data structure that contains a collection of elements.
* Stack implements the LIFO mechanism i.e. the element that is pushed at the end is popped out first.

Some of the principle operations in the stack are −

* **Push** - This adds a data value to the top of the stack.
* **Pop** - This removes the data value on top of the stack.
* **Peek** - This returns the top data value of the stack.

A program that implements a stack using linked list is given as follows.

## Example

#include <iostream>

#include <cstdlib>

using namespace std;

struct Node

{

   int data;

   struct Node \*next;

};

struct Node\* top = NULL;

void push(int val)

{

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

   newnode->data = val;

   newnode->next = top;

   top = newnode;

}

void pop()

{

   if(top==NULL)

   cout<<"Stack Underflow"<<endl;

   else {

      cout<<"The popped element is "<< top->data <<endl;

      top = top->next;

   }

}

void display()

{

   struct Node\* ptr;

   if(top==NULL)

   cout<<"stack is empty";

   else {

      ptr = top;

      cout<<"Stack elements are: ";

      while (ptr != NULL) {

         cout<< ptr->data <<" ";

         ptr = ptr->next;

      }

   }

   cout<<endl;

}

int main()

{

   int ch, val;

   cout<<"1) Push in stack"<<endl;

   cout<<"2) Pop from stack"<<endl;

   cout<<"3) Display stack"<<endl;

   cout<<"4) Exit"<<endl;

   do {

      cout<<"Enter choice: "<<endl;

      cin>>ch;

      switch(ch)

{

         case 1: {

            cout<<"Enter value to be pushed:"<<endl;

            cin>>val;

            push(val);

            break;

         }

         case 2: {

            pop();

            break;

         }

         case 3: {

            display();

            break;

         }

         case 4: {

            cout<<"Exit"<<endl;

            break;

         }

         default: {

            cout<<"Invalid Choice"<<endl;

         }

      }

   }while(ch!=4);

   return 0;

}

In the above program, the structure Node is used to create the linked list that is implemented as a stack. The code is given below.

struct Node {

int data;

struct Node \*next;

};

The push() function takes argument val i.e. value to be pushed into the stack. Then a new node is created and val is inserted into the data part. This node is added to the front of the linked list and top points to it. The code snippet for this is as follows.

void push(int val) {

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

   newnode->data = val;

   newnode->next = top;

   top = newnode;

}

The pop() function pops the topmost value of the stack, if there is any value. If the stack is empty then underflow is printed. This is given as follows.

void pop() {

   if(top==NULL)

   cout<<"Stack Underflow"<<endl;

   else {

      cout<<"The popped element is "<< top->data <<endl;

      top = top->next;

   }

}

The display() function displays all the elements in the stack. This is done by using ptr that initially points to top but goes till the end of the stack. All the data values corresponding ti ptr are printed. This is given below.

void display() {

   struct Node\* ptr;

   if(top==NULL)

   cout<<"stack is empty";

   else {

      ptr = top;

      cout<<"Stack elements are: ";

      while (ptr != NULL) {

         cout<< ptr->data <<" ";

         ptr = ptr->next;

      }

   }

   cout<<endl;

}

The function main() provides a choice to the user if they want to push, pop or display the stack. According to the user response, the appropriate function is called using switch. If the user enters an invalid response, then that is printed. The code snippet for this is given below.

int main() {

   int ch, val;

   cout<<"1) Push in stack"<<endl;

   cout<<"2) Pop from stack"<<endl;

   cout<<"3) Display stack"<<endl;

   cout<<"4) Exit"<<endl;

   do {

      cout<<"Enter choice: "<<endl;

      cin>>ch;

      switch(ch) {

         case 1: {

            cout<<"Enter value to be pushed:"<<endl;

            cin>>val;

            push(val);

            break;

         }

         case 2: {

            pop();

            break;

         }

         case 3: {

            display();

            break;

         }

         case 4: {

            cout<<"Exit"<<endl;

            break;

         }

         default: {

            cout<<"Invalid Choice"<<endl;

         }

      }

   }while(ch!=4);

   return 0;

}