Implementation of Stack using Array

```
#include <iostream>
using namespace std;
int stack[100], n=100, top=-1;
void push(int val) {
  if(top>=n-1)
  cout<<"Stack Overflow"<<endl;
  else {
    top++;
    stack[top]=val;
  }
}
void pop() {
  if(top<=-1)
  cout<<"Stack Underflow"<<endl;
  else {
    cout<<"The popped element is "<< stack[top] <<endl;
    top--;
  }
}
void display() {
  if(top>=0) {
    cout<<"Stack elements are:";
    for(int i=top; i>=0; i--)
    cout<<stack[i]<<" ";
    cout<<endl;
  } else
```

```
  cout<<"Stack is empty";
}
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;
}
```

Implementation of Stack using Linked List

```
#include <iostream>
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
```