#include <stdio.h>

#include <stdlib.h>

// Structure to create a node with data and the next pointer

struct Node {

int data;

struct Node \*next;

};

Node\* top = NULL;

// Push() operation on a stack

void push(int value) {

struct Node \*newNode;

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

newNode->data = value; // assign value to the node

if (top == NULL) {

newNode->next = NULL;

} else {

newNode->next = top; // Make the node as top

}

top = newNode; // top always points to the newly created node

printf("Node is Inserted\n\n");

}

int pop() {

if (top == NULL) {

printf("\nStack Underflow\n");

} else {

struct Node \*temp = top;

int temp\_data = top->data;

top = top->next;

free(temp);

return temp\_data;

}

}

void display() {

// Display the elements of the stack

if (top == NULL) {

printf("\nStack Underflow\n");

} else {

printf("The stack is \n");

struct Node \*temp = top;

while (temp->next != NULL) {

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

temp = temp->next;

}

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

}

}

int main() {

int choice, value;

printf("\nImplementation of 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("\nWrong Choice\n");

}

}

}