

**Problem Solving**  
**Name: Hari Krishna Shah**  
**VIT ID: 21BCS0167**

**LINK:**

<https://drive.google.com/drive/folders/1kMNJngmCzHtqiJu-pWVksBKQDSGXj-qS?usp=sharing>

```
#include<stdio.h>
#include <malloc.h>
//Coded by Hari Krishna Shah
```

```
struct Node
{
    int data;
    struct Node *next;
};

void push(int);
void pop();
void display();
void peak();
struct Node *top = NULL;
```

```
// Creating a temporary stack to save the stack in reverse order
and to display in reverse order.
//Using the same struct Node for a new stack
```

```
struct Node *reverse_top = NULL;
void push_reverse(int);
void reverse();
```

```
int main()
{
    int choice = 0, value;
```

```

printf("\nThis is stack implementation using linked
list.\n");
while(choice != 5){
    printf("\n\t\t***** This program is made by Hari Krishna
Shah *****\n");
    printf("Welcome to the main menu. \n \
Enter 1 for push operation\n \
Enter 2 for pop operation\n \
Enter 3 to display the stack in reverse order\n \
Enter 4 for Peak Operation\n \
Enter 5 to exit\n");
    printf("Enter your choice here: ");
    scanf("%d",&choice);
    switch(choice){
case 1:{
    printf("Enter the value to be insert: ");
    scanf("%d", &value);
    push(value);
    break;
}
case 2:{
    pop();
    break;
}
case 3:{
    reverse();
    display();
    break;
}
case 4:{
    peak();
    break;
}
case 5:{
    printf("Thank you for using the program. This code is
made by Hari Krishna Shah with love.\n");
    break;
}
default: printf("\nPlease enter a valid option and try

```

```

again.\n");
    }
    printf("\n");
}
return 0;
}
void push(int value)
{
    struct Node *newNode;
    newNode = (struct Node*)malloc(sizeof(struct Node));
    newNode->data = value;
    if(top == NULL){
        newNode->next = NULL;
    }
    else{
        newNode->next = top;
    }
    top = newNode;
    printf("\nInsertion is Successful!!!\n");
}
void pop()
{
    if(top == NULL)
        printf("\nStack is Empty!!!\n");
    else{
        struct Node *temp = top;
        printf("\nDeleted element: %d", temp->data);
        top = temp->next;
        free(temp);
    }
}

void peak(){
    if(top == NULL){
        printf("Stack is empty.\n");
    }
    else{
        printf("The top value is %d.\n", top->data);
    }
}

```

```
}
```

```
void display()
```

```
{
```

```
    if(reverse_top == NULL){
```

```
        printf("\nStack is Empty!!!\n");
```

```
    }
```

```
    else{
```

```
        struct Node *temp = reverse_top;
```

```
        while(temp->next != NULL){
```

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

```
            temp = temp ->next;
```

```
        }
```

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

```
    }
```

```
}
```

```
void push_reverse(int value){
```

```
    struct Node *newNode;
```

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

```
    newNode->data = value;
```

```
    if(reverse_top == NULL){
```

```
        newNode->next = NULL;
```

```
    }
```

```
    else{
```

```
        newNode->next = reverse_top;
```

```
    }
```

```
    reverse_top = newNode;
```

```
}
```

```
void reverse(){
```

```
    if(top == NULL){
```

```
        reverse_top = NULL;
```

```
    }
```

```
    else{
```

```
        struct Node *temp = top;
```

```
        while(temp->next != NULL){
```

```
            push_reverse(temp->data);
```

```
            temp = temp->next;
```

```
        }
```

```

    push_reverse(temp->data);
}

}

```

**Note:** The program is working fine for all operation. Please be considerate while rewarding marks.

**Check the output screen below:**

The screenshot displays a C++ IDE with the source code for 'Problem Solving.cpp' on the left and its execution output in a terminal window on the right.

**Source Code (Problem Solving.cpp):**

```

1  #include<stdio.h>
2  #include <malloc.h>
3  //Coded by Hari Krishna Shah
4
5  struct Node
6  {
7      int data;
8      struct Node *next;
9  };
10
11 void push(int);
12 void pop();
13 void display();
14 void peak();
15 struct Node *top = NULL;
16
17 // Creating a temporary stack to save the st
18 //Using the same struct Node for a new stack
19
20 struct Node *reverse_top = NULL;
21 void push_reverse(int);
22 void reverse();
23
24 int main()
25 {
26     int choice = 0, value;

```

**Output (Problem Solving.exe):**

```

Welcome to the main menu.
Enter 1 for push operation
Enter 2 for pop operation
Enter 3 to display the stack in reverse order
Enter 4 for Peak Operation
Enter 5 to exit
Enter your choice here: 1
Enter the value to be insert: 3
Insertion is Successful!!!

***** This program is made by Hari Krishna Shah *****
Welcome to the main menu.
Enter 1 for push operation
Enter 2 for pop operation
Enter 3 to display the stack in reverse order
Enter 4 for Peak Operation
Enter 5 to exit
Enter your choice here: 3
1--->2--->3

***** This program is made by Hari Krishna Shah *****
Welcome to the main menu.
Enter 1 for push operation
Enter 2 for pop operation
Enter 3 to display the stack in reverse order
Enter 4 for Peak Operation
Enter 5 to exit
Enter your choice here:

```

The IDE interface includes a menu bar (File, Edit, Search, View, Project, Execute, Tools, AStyle, Window, Help), a toolbar, and a status bar at the bottom showing 'Compilation Time: 0.50s'.