ASSIGNMENT NO : 3

PROGRAM:

```
#include <iostream>
#include <stdlib.h>
#include <string.h>
#include<ctype.h>
using namespace std;
struct node {
int data;
struct node *next;
};
struct node *top = NULL;
/* create a new node with the given data */
struct node* createNode(int data)
{
struct node *ptr = (struct node *) malloc(sizeof (struct node));
ptr->data = data;
ptr->next = NULL;
}

void push (int data) {
struct node *ptr= createNode(data);
if (top == NULL) {
top = ptr;
```

```
return;
}
ptr->next = top;
top = ptr;
}
/* pop the top element from the stack */
int pop () {
int data;
struct node *temp;
if (top == NULL)
return -1;
data= top->data;
temp = top;
top = top->next;
free(temp);
return (data);
}
int main() {
char str[100];
int i, data=-1, operand1, operand2, result;
/* i/p postfix expr from the user */
cout <<"Enter ur postfix expression:";
fgets(str, 100, stdin);
for (i= 0;i< strlen(str); i++){
```

```
if (isdigit(str[i])){
/** if the i/p char is digit, parse * character by character to get * complete operand
*/
data= (data ==-1)?0: data;

data = (data * 10) + (str[i]-48);
continue;
}
/* push the operator into the stack */
if (data !=-1){
push(data);
}
if (str[i] == '+' || str[i] =='-'
|| str[i] == '*' || str[i] == '/'){
/*
 * if the i/p character is an operator,
 * then pop two elements from the stack,
 * apply operator and push the result into
 * the stack
*/
operand2= pop();
operand1= pop();
if (operand1 ==-1 || operand2 ==-1)
```

```
break;
switch (str[i]) {
case '+':
result= operand1+ operand2;
/* pushing result into the stack */
push(result);
break;
case '-':
result = operand1 - operand2;
push(result);
break;
case '*':
result = operand1 * operand2;
push(result);
break;
case '/':
result = operand1 / operand2;
push(result);

break;
}
}
data = -1;
}
```

```
if (top != NULL && top->next == NULL)
cout<<"Output:"<< top->data;
else
cout<<"u ve entered wrong expression\n";
return 0;
}
```

OUTPUT :