

```

1  /*
2  TITLE: Evaluation of Postfix Expression
3  NAME:Tauseef Mushtaque Ali Shaikh
4  CLASS: S.Y.[CO]
5  ROLLNO: 18CO63
6  SUBJECT: DS
7  DATE: 19/8/19
8  DISCRIPTION: In this Program a Postfix Expression is Evaluated and Result is
9  Stored in a Stack.
10 */
11 #include<stdio.h>
12 #include<stdlib.h>
13 #define max 50
14
15 struct stack
16 {
17     int data[max];
18     int top;
19 };
20
21 int empty(struct stack *s)
22 {
23     return (s->top== -1)?1:0;
24 }
25
26 void push(struct stack *s,int ele)
27 {
28     if (s->top < max-1)
29         s->data[++s->top]=ele;
30     else
31         printf("\nSTACK OVERFLOW");
32 }
33
34 int pop(struct stack *s)
35 {
36     if(!empty(s))
37         return s->data[s->top--];
38     else
39         return (int)-1;
40 }
41
42 int eval(char *expr)
43 {
44     char c;
45     int i,res,op2;
46     struct stack st;
47     st.top=-1;
48     for(i=0;expr[i]!='\0';i++)
49     {
50         c=expr[i];
51         switch(c)
52         {
53             case '+':
54                 op2=pop(&st);
55                 res=pop(&st)+op2;

```

```

56         push(&st,res);
57         break;
58
59         case '-':
60             op2=pop(&st);
61             res=pop(&st)-op2;
62             push(&st,res);
63             break;
64
65         case '*':
66             op2=pop(&st);
67             res=pop(&st)*op2;
68             push(&st,res);
69             break;
70
71         case '/':
72             op2=pop(&st);
73             res=pop(&st)/op2;
74             push(&st,res);
75             break;
76
77         case '%':
78             op2=pop(&st);
79             res=pop(&st)%op2;
80             push(&st,res);
81             break;
82
83
84         case '$':
85             op2=pop(&st);
86             res=pow(pop(&st),op2);
87             push(&st,res);
88             break;
89
90         case '0':
91             exit(0);
92             break;
93
94         default:
95             push(&st,c-'0');
96
97     }
98 }
99 return pop(&st);
100 }
101
102 int main()
103 {
104     while(1)
105     {
106         char *postfix;
107         int res;
108         postfix=(char*)malloc(1);
109         printf("\nEnter postfix Expression:");
110         scanf("%s",postfix);
111         res=eval(postfix);

```

```
112         printf("\n Evaluation to Postfix Expression is : %d\n",res);
113         printf("\nEnter 0 to exit the program\n");
114     }
115 }
116
117
118
```