

Practical No. 08

Name: Sushrut Rajesh Babhulkar

PRN: 1841003

Batch: B1

Class: L.Y. B.Tech. (Computer Engineering)

Subject: CO406UC Compiler Design Lab

Aim: Write a C program to implement operator precedence parsing.

Program:

```
#include<stdio.h>

#include<stdlib.h>

#include<string.h>

int main()

{

    char stack[20],ip[20],opt[10][10][1],ter[10];

    int i,j,k,n,top=0,col,row;

    for(i=0;i<10;i++)

    {

        stack[i]=NULL;

        ip[i]=NULL;

        for(j=0;j<10;j++)
```

```

        {
            opt[i][j][1]=NULL;
        }
    }

    printf("Enter the no.of terminals:\n");
    scanf("%d",&n);

    printf("\nEnter the terminals:\n");
    for(i=0;i<n;i++)
    {
        scanf("%s",&ter[i]);
    }

    printf("\nEnter the table values:\n");
    for(i=0;i<n;i++)
    {
        for(j=0;j<n;j++)
        {
            printf("Enter the value for %c %c: ",ter[i],ter[j]);
            scanf("%s",opt[i][j]);
        }
    }

    printf("\n** OPERATOR PRECEDENCE TABLE **\n");
    for(i=0;i<n;i++)
    {
        printf("\t%c",ter[i]);
    }

```

```

}

printf("\n");

for(i=0;i<n;i++)

{

    printf("\n%c",ter[i]);

    for(j=0;j<n;j++)

    {

        printf("\t%c",opt[i][j][0]);

    }

}

stack[top]='$';

printf("\nEnter the input string: ");

scanf("%s",ip);

i=0;

printf("\nSTACK\t\t\tINPUT STRING\t\t\tACTION\n");

printf("\n%s\t\t\t%s\t\t\t",stack,ip);

while(i<=strlen(ip))

{

    for(k=0;k<n;k++)

    {

        if(stack[top]==ter[k])

            col=k;

        if(ip[i]==ter[k])

            row=k;

```

```

    }

    if((stack[top]=='$')&&(ip[i]=='$'))
    {
        printf("\nString is accepted\n");
        break;
    }
    else if((opt[col][row][0]=='<') || (opt[col][row][0]=='='))
    {
        stack[++top]=opt[col][row][0];
        stack[++top]=ip[i];
        printf("Shift %c",ip[i]);
        i++;
    }
    else
    {
        if(opt[col][row][0]=='>')
        {
            while(stack[top]!='<'){--top;}
            top=top-1;
            printf("Reduce");
        }
    }
    else
    {
        printf("\nString is not accepted");
    }

```

```
        break;
    }
}
printf("\n");
for(k=0;k<=top;k++)
{
    printf("%c",stack[k]);
}
printf("\t\t\t");
for(k=i;k<strlen(ip);k++)
{
    printf("%c",ip[k]);
}
printf("\t\t\t");
}
}
```

Output:

```
G:\V Comp Engineering\CDL\OperatorPrecedenceParser.exe
Enter the no.of terminals:
4
Enter the terminals:
+
*
a
$
Enter the table values:
Enter the value for + +: >
Enter the value for + *: <
Enter the value for + a: <
Enter the value for + $: >
Enter the value for * +: >
Enter the value for * *: >
Enter the value for * a: <
Enter the value for * $: >
Enter the value for a +: >
Enter the value for a *: >
Enter the value for a a: =
Enter the value for a $: >
Enter the value for $ +: <
Enter the value for $ *: <
Enter the value for $ a: <
Enter the value for $ $: A

** OPERATOR PRECEDENCE TABLE **
+      *      a      $
+      >      <      <      >
*      >      >      <      >
a      >      >      =      >
$      <      <      <      A
Enter the input string: a+a*a$
```

```
G:\V Comp Engineering\CDL\B01\OperatorPrecedenceParser.exe
** OPERATOR PRECEDENCE TABLE **
+      *      a      $
+      >      <      <      >
*      >      >      <      >
a      >      >      =      >
$      <      <      <      A
Enter the input string: a+a*a$

STACK      INPUT STRING      ACTION
$          a+a*a$a      Shift a
$<a        +a*a$a      Reduce
$          +a*a$a      Shift +
$<+        a*a$a      Shift a
$<+<a      *a$a      Reduce
$<+<a      *a$a      Shift *
$<+<a*     a$a      Shift a
$<+<a*a    $      Reduce
$<+<a*a    $      Reduce
$<+<a*a    $      Reduce
$          $          String is accepted

-----
Process exited after 119.6 seconds with return value 0
Press any key to continue . . .
```