Addi Akash
1BM19CS007

Date

Expt. No. 2


```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#define SIZE 20

Char stack[SIZE];
int top = -1;
void push(char ele)
{ if (top >= SIZE)
   { printf ("\n Stack overflow");
   }
   else { top = top+1;
          Stack[top] = ele;
        } }
Char pop()
{
  Char ele;
  if (top == -1)
{ printf("Stack underflow");
  getchar();
  exit(1); }
else { ele = stack[top];
       top = top - 1;
       return(ele); } }

int is_opt (char symbol)
{ if (symbol == '^' || symbol == '*' ||
     symbol == '/' || symbol == '+' || symbol == '-')
 { return 1; }
```

Teacher's Signature

```
else {
    return 0; } }

int higher (char symbol)
{
 switch(symbol)
{ case '^': return(3);
            break;
 Case '*':
 Case '/':
            return(2);
            break;
 case '+':
 case '-': return(1);
            break;
 default : return(0);
            break;
   } }
void infintopostfin(char infin_exp[], char
        postfin_exp[])
{
  int i=0, j=0;
  char ele;
  char x;
  push('(');
  strcat(infin_exp,")");
  ele = infin_exp[i];
  while (ele != '\0')
  {
```

Aditi Akarsh
1BM19CS007

Date

Expt. No. 2


```
if (ele == '(')
{ push (ele); }
else if (ele == 'A' || ele == 'B' || ele == 'C' || ele == 'D' ||
         ele == '1' || ele == '2' || ele == '3' || ele == '9')
{ postfin-exp[j] = ele;
    j++;
}
else if (is_operator (ele) == 1)
 { x = pop ();
   while (is_operator(x) == 1 && higher (x) >=
   higher(ele)) {
                postfin-exp[j] = x;
                  j++;
                x = pop(); }
   push(x);
   push(ele); }
else if (ele == ')')
 { x = pop();
   while (x != '(')
   { postfin-exp[j] = x;
     j++;
     x = pop(); } }
 else
   {
     printf ("Invalid infin Expression");
      getchar();
      exit(1);
}
 i++;
```

Teacher's Signature

Aditi Akash
1BM19CS007

Date 6

Expt. No. 2

Page No.

```
ele = infin_exp[i];
}
  postfin_exp[j]='\0'; }

int main()
{
    char infin[SIZE], postfin[SIZE];
    printf("Enter infix expression ");
    gets(infin);
    infintopostfin(infin, postfin);
    printf("Postfin expression:")
    puts(postfin);
    return 0;
}
```

Teacher's Signature