S.No: 39	Exp. Name: ***Infix to Prefix Conversion***	Date:2023-06-19

Aim:

Write a C program to convert an Infix expression to Prefix expression.

Source Code:

infixToPrefix.c

```
#include<stdio.h>
#include<string.h>
#include<ctype.h>
#define SIZE 50
char *strrev(char *str){
   char c, *front, *back;
   if(!str || !*str) {
      return str;
   }
   for(front=str,back=str+strlen(str)-1;front < back;front++,back--) {
      c=*front;*front=*back;*back=c;
   }
   return str;
}
char s[SIZE];
int top = -1;
void push(char ele) {
   s[++top] = ele;
}
char pop() {
   return(s[top--]);
}
int pr(char ele) {
   switch(ele) {
      case '#':
      return 0;
      case ')':
      return 1;
      case '+':
      case '-':
      return 2;
      case '/':
      return 3;
   }
}
void main() {
   char infx[50],prfx[50], ch, elem;
   int i = 0,k = 0;
   printf("Enter Infix Expression:");
   scanf("%s",infx);
   push('#');
   strrev (infx);
   while((ch = infx[i++]) !='\0') {
      if(ch==')')
      push(ch);
      else if( isalnum (ch))
```


ID: 224G1A05B6
2022-2026-CSE-B
Srinivasa Ramanujan Institute of Technology

```
      prfx[k++] = ch;
      else if(ch=='(') {
         while(s[top]!='(') {
            prfx[k++]= pop();
         }
         elem = pop();
        }
   else {
      while(pr (s[top]) >= pr (ch)) {
         prfx[k++] = pop();
      }
      push(ch);
   }
}
while (s[top] != '#') {
   prfx[k++] = pop();
}
prfx[k] = '\0';
strrev(prfx);
strrev(infx);
printf("Prefix Expression:%s\n",prfx);
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Enter Infix Expression: A+B
Prefix Expression:+AB

Test Case - 2
User Output
Enter Infix Expression: A/B+C/D
Prefix Expression:+/AB/CD