

IN-FIX TO POST-FIX

CODE:

```
#include <stdio.h>
#include<ctype.h>
#define SIZE 100
char stack[SIZE];
int top=-1;
void push(char ch){
    stack[++top]=ch;
}
char pop(){
    return stack[top--];
}
char peek(){
    return stack[top];
}
int precedence(char op){
    if(op=='+'||op=='-'){
        return 1;
    }
    if(op=='*'||op=='/'){
        return 2;
    }
    return 0;
}
void intopost(char* infix){
    char ch,postfix[SIZE];
    int i=0,j=0;
    while((ch = infix[i++]) != '\0'){
```

```

        if(isalnum(ch)){
            postfix[j++]=ch;
        }
        else if(ch=='('){
            push(ch);
        }
        else if(ch==')'){
            while(peek()!='(')
                postfix[j++]=pop();
            pop();
        }
        else{
            while(top!=-1&&precedence(peek())>=precedence(ch))
                postfix[j++]=pop();
            push(ch);
        }
    }

    while(top != -1){
        postfix[j++] = pop();
    }

    postfix[j] = '\0';

    printf("Postfix Expression: %s\n", postfix);
}

int main(){
    char infix[SIZE];

    printf("Enter Infix expression:");

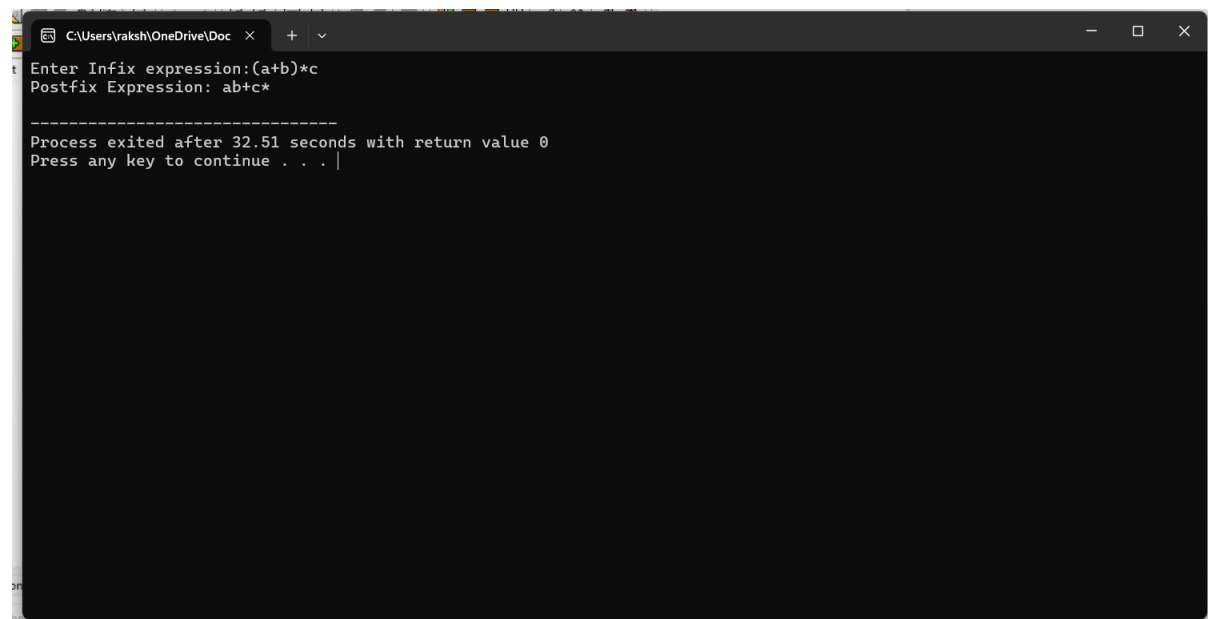
    scanf("%s",infix);

    intopost(infix);

    return 0;
}

```

OUTPUT:



```
C:\Users\raksh\OneDrive\Doc > Enter Infix expression:(a+b)*c
Postfix Expression: ab+c*

-----
Process exited after 32.51 seconds with return value 0
Press any key to continue . . .
```