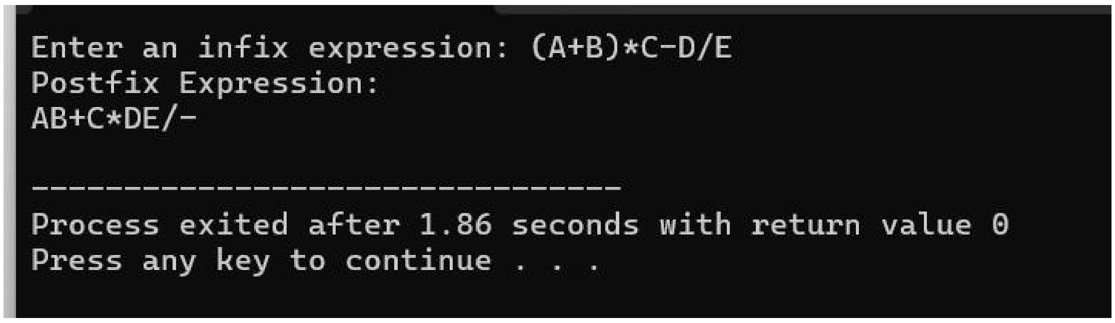
INFIX TO POSTFIX

```
#include<stdio.h>
#include<ctype.h>
#include<stdlib.h>
#include<string.h>
#define MAX 100
char stack[MAX];
int top=-1;
void push(char c){
        if(top==MAX-1){
                printf("Stackoverflow\n");
        }
        else{
                stack[++top]=c;
        }
}
char pop(){
        if (top==0){
                return -1;
        }
        else{
                return stack[top--];
        }
}
int precedence(char c) {
    if (c=='^')
        return 3;
    else if(c=='*'||c=='/')
        return 2;
    else if (c=='+'||c=='-')
        return 1;
    else
        return 0;
}
void infixtopostfix(char infix[]){
        char postfix[MAX];
    int i,k=0;
    char ch;
    printf("Postfix Expression:\n");
    for(i=0;infix[i]!='\0';i++){
        ch=infix[i];
        if(isalnum(ch)){
                printf("%c",ch);
                }
                else if(ch=='('){
                        push(ch);
                }
                else if(ch==')'){
                        while((ch=pop())!='(')
                                printf("%c",ch);
                }
                else{
                        while(top!=-1&&precedence(stack[top])>=precedence(ch))
                                printf("%c",pop());
                        push(ch);
                }
        }
        while(top!=-1)
                printf("%c",pop());
        printf("\n");
}
int main() {
   char infix[MAX];
   printf("Enter an infix expression: ");
   gets(infix);
   infixtopostfix(infix);
   return 0;
}
```

OUTPUT:



```
Enter an infix expression: (A+B)*C-D/E
Postfix Expression:
AB+C*DE/-

--------------------------------
Process exited after 1.86 seconds with return value 0
Press any key to continue . . .
```