

```

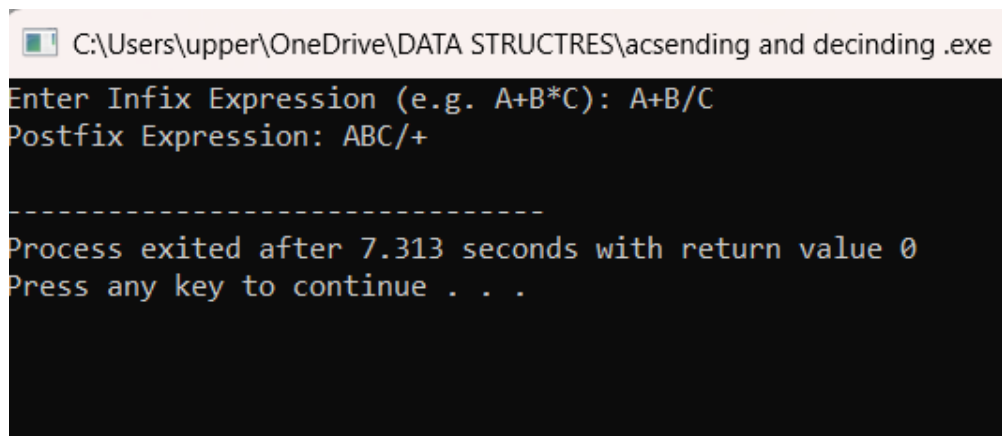
#include <stdio.h>
#include <ctype.h>
#include <string.h>
#define SIZE 100
char stack[SIZE];
int top = -1;
void push(char c) {
    stack[++top] = c;
}
char pop() {
    if(top == -1)
        return -1;
    else
        return stack[top--];
}
char peek() {
    return stack[top];
}
int precedence(char op) {
    if(op == '^') return 3;
    if(op == '*' || op == '/') return 2;
    if(op == '+' || op == '-') return 1;
    return 0;
}
void infixToPostfix(char infix[]) {
    char postfix[SIZE];
    int i = 0, j = 0;
    char ch;
    while(infix[i] != '\0') {
        ch = infix[i];
        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);
}
i++;
}
while(top != -1) {
    postfix[j++] = pop();
}
postfix[j] = '\0';
printf("Postfix Expression: %s\n", postfix);
}
int main() {
    char infix[SIZE];
    printf("Enter Infix Expression (e.g. A+B*C): ");
    scanf("%s", infix);
    infixToPostfix(infix);
    return 0;
}

```



```

C:\Users\upper\OneDrive\DATA STRUCTRES\ascending and decinding .exe
Enter Infix Expression (e.g. A+B*C): A+B/C
Postfix Expression: ABC/+

-----
Process exited after 7.313 seconds with return value 0
Press any key to continue . . .

```