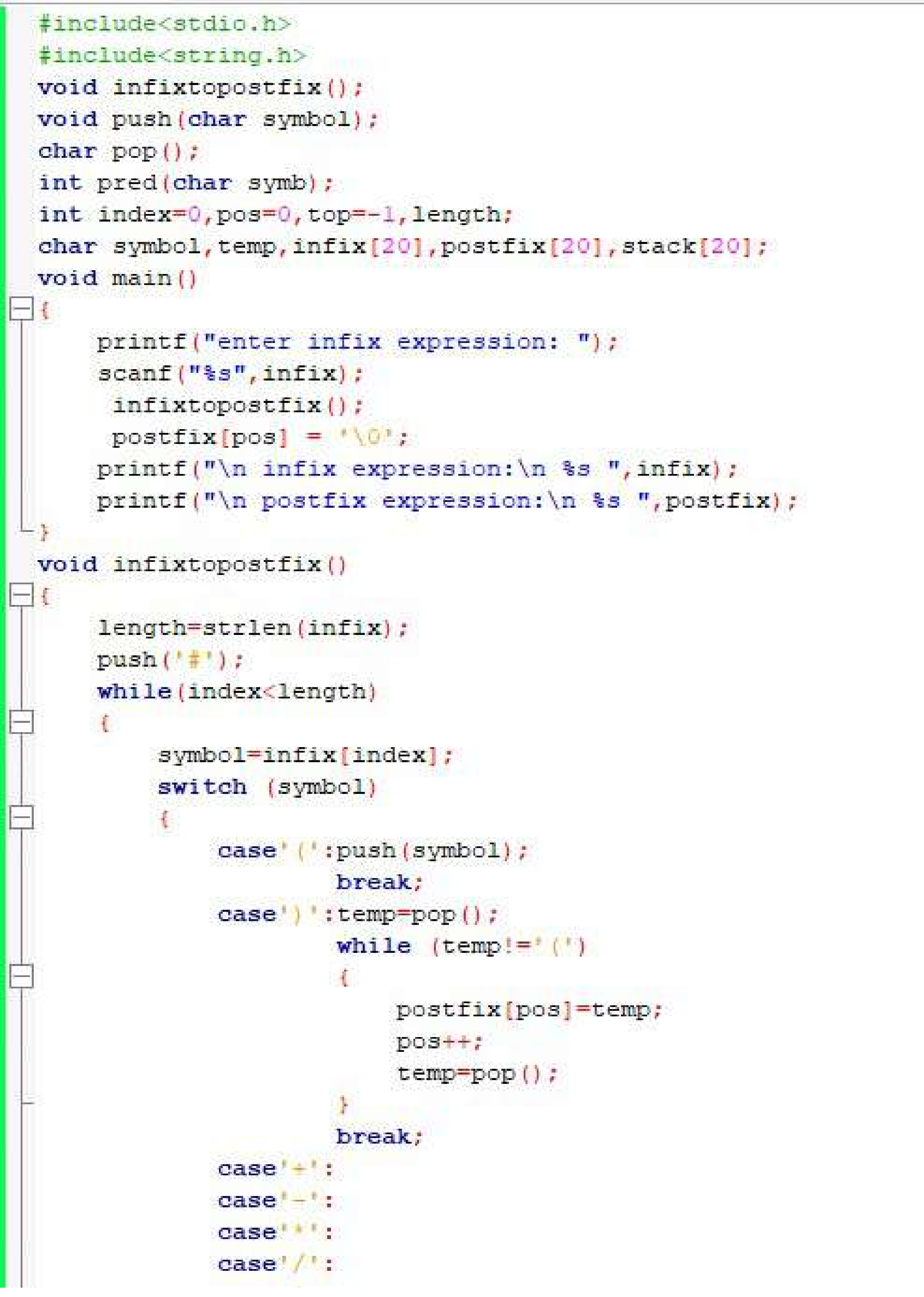
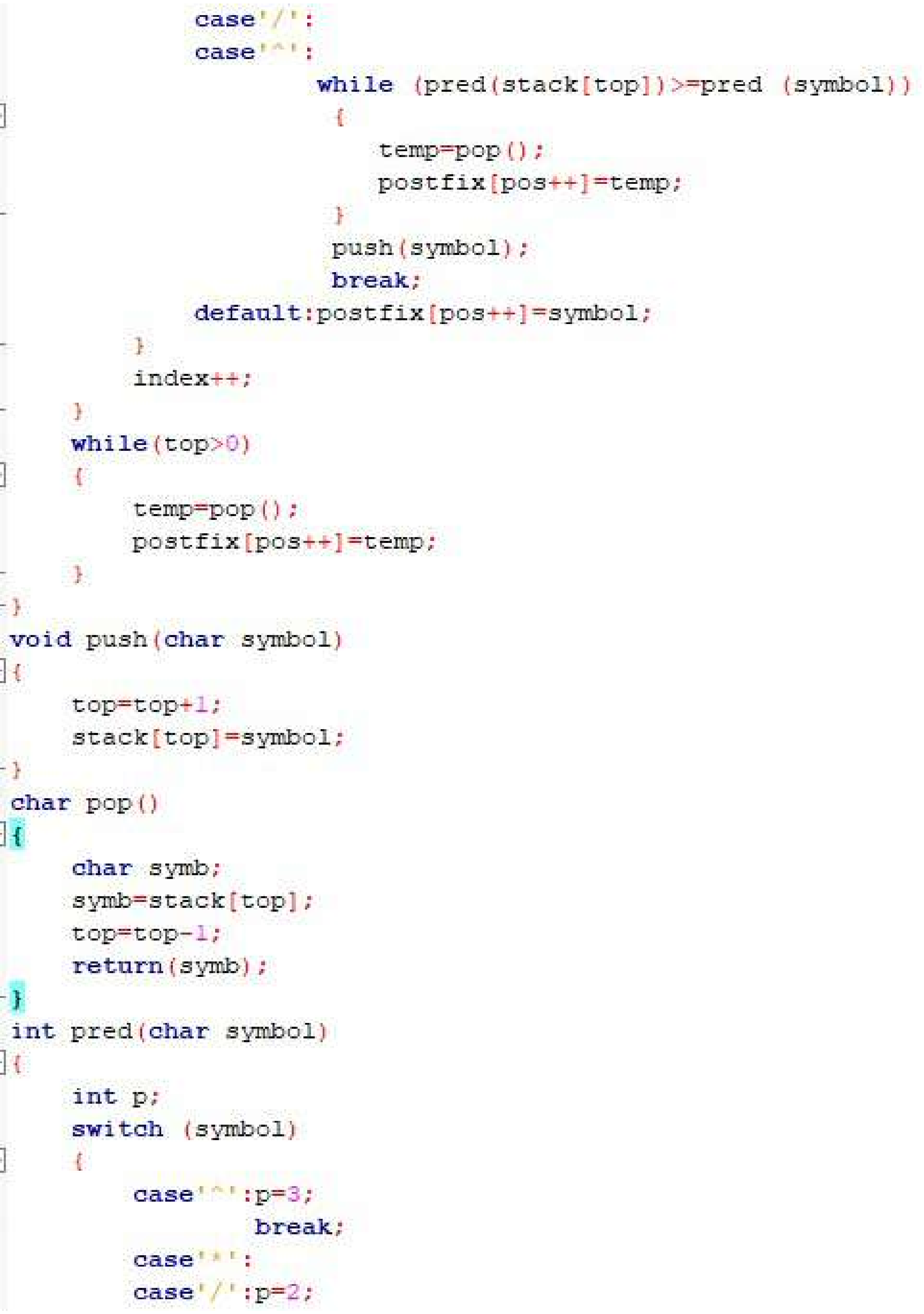
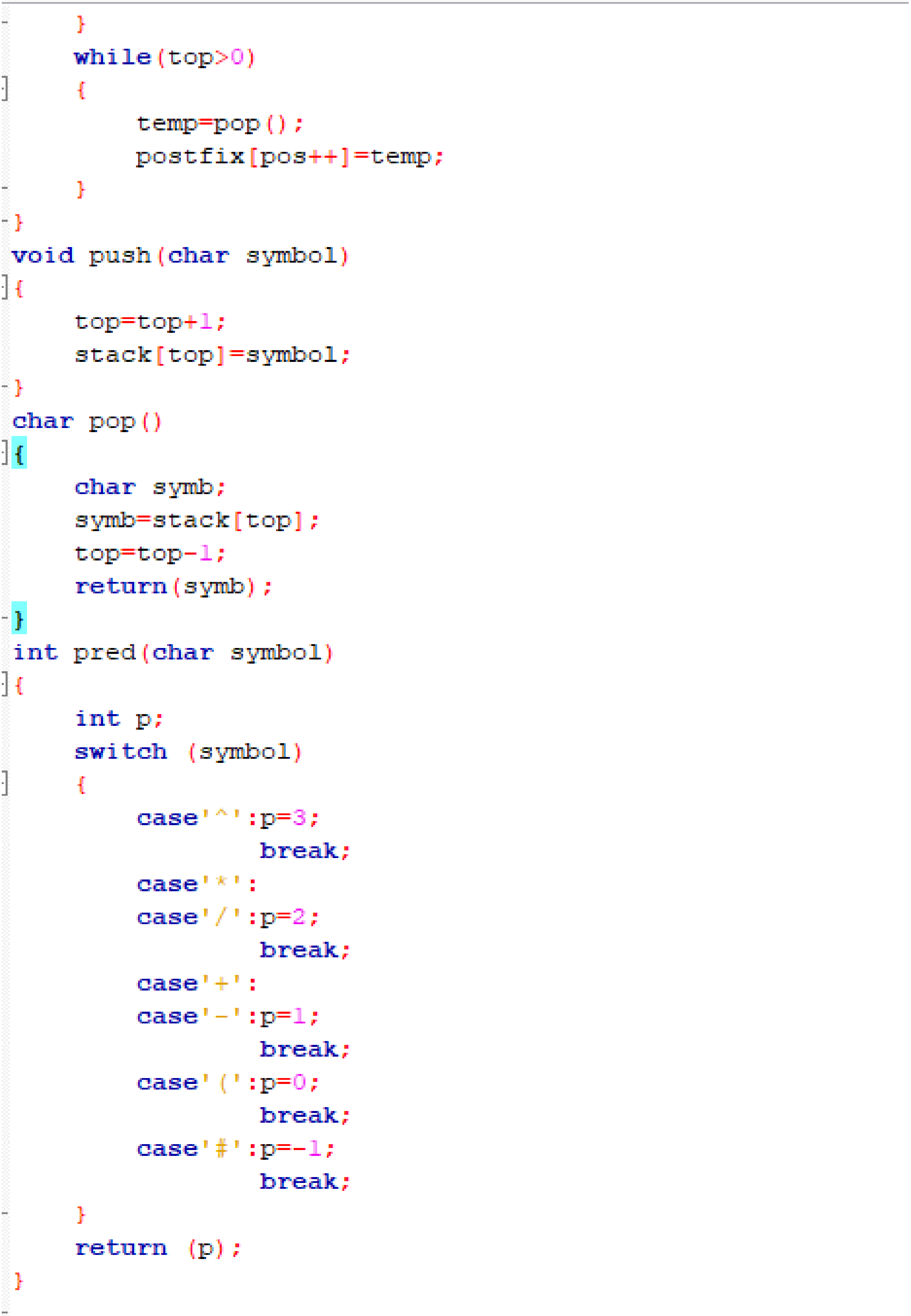


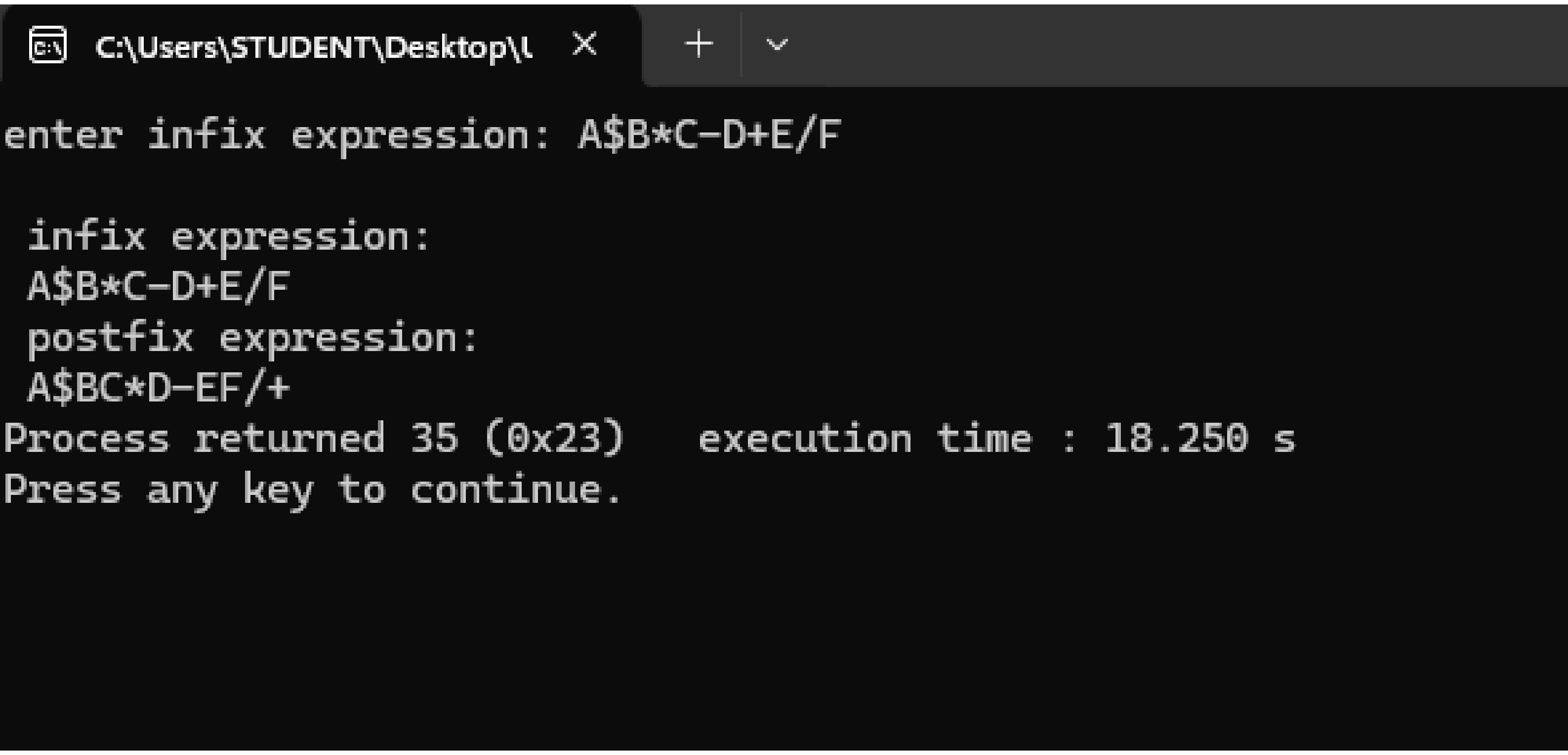
```
#include<stdio.h>
#include<string.h>
void infixtopostfix();
void push(char symbol);
char pop();
int pred(char symb);
int index=0,pos=0,top=-1,length;
char symbol,temp,infix[20],postfix[20],stack[20];
void main()
{
    printf("enter infix expression: ");
    scanf("%s",infix);
     infixtopostfix();
     postfix[pos] = '\0';
    printf("\n infix expression:\n %s ",infix);
    printf("\n postfix expression:\n %s ",postfix);
}
void infixtopostfix()
{
    length=strlen(infix);
    push('#');
    while(index<length)
    {
        symbol=infix[index];
        switch (symbol)
        {
            case'(':push(symbol);
                    break;
            case')':temp=pop();
                    while (temp!='(')
                    {
                        postfix[pos]=temp;
                        pos++;
                        temp=pop();
                    }
                    break;
            case'+':
            case'-':
            case'*':
            case'/':
```



```
            case'/':
            case'^':
                    while (pred(stack[top])>=pred (symbol))
                     {
                        temp=pop();
                        postfix[pos++]=temp;
                     }
                     push(symbol);
                     break;
            default:postfix[pos++]=symbol;
        }
        index++;
    }
    while(top>0)
    {
        temp=pop();
        postfix[pos++]=temp;
    }
}
void push(char symbol)
{
    top=top+1;
    stack[top]=symbol;
}
char pop()
{
    char symb;
    symb=stack[top];
    top=top-1;
    return(symb);
}
int pred(char symbol)
{
    int p;
    switch (symbol)
    {
        case'^':p=3;
                break;
        case'*':
        case'/':p=2;
```



```
    }
    while(top>0)
    {
        temp=pop();
        postfix[pos++]=temp;
    }
}
void push(char symbol)
{
    top=top+1;
    stack[top]=symbol;
}
char pop()
{
    char symb;
    symb=stack[top];
    top=top-1;
    return(symb);
}
int pred(char symbol)
{
    int p;
    switch (symbol)
    {
        case'^':p=3;
                break;
        case'*':
        case'/':p=2;
                break;
        case'+':
        case'-':p=1;
                break;
        case'(':p=0;
                break;
        case'#':p=-1;
                break;
    }
    return (p);
}
```



```
enter infix expression: A$B*C-D+E/F

 infix expression:
 A$B*C-D+E/F
 postfix expression:
 A$BC*D-EF/+
Process returned 35 (0x23)   execution time : 18.250 s
Press any key to continue.
```