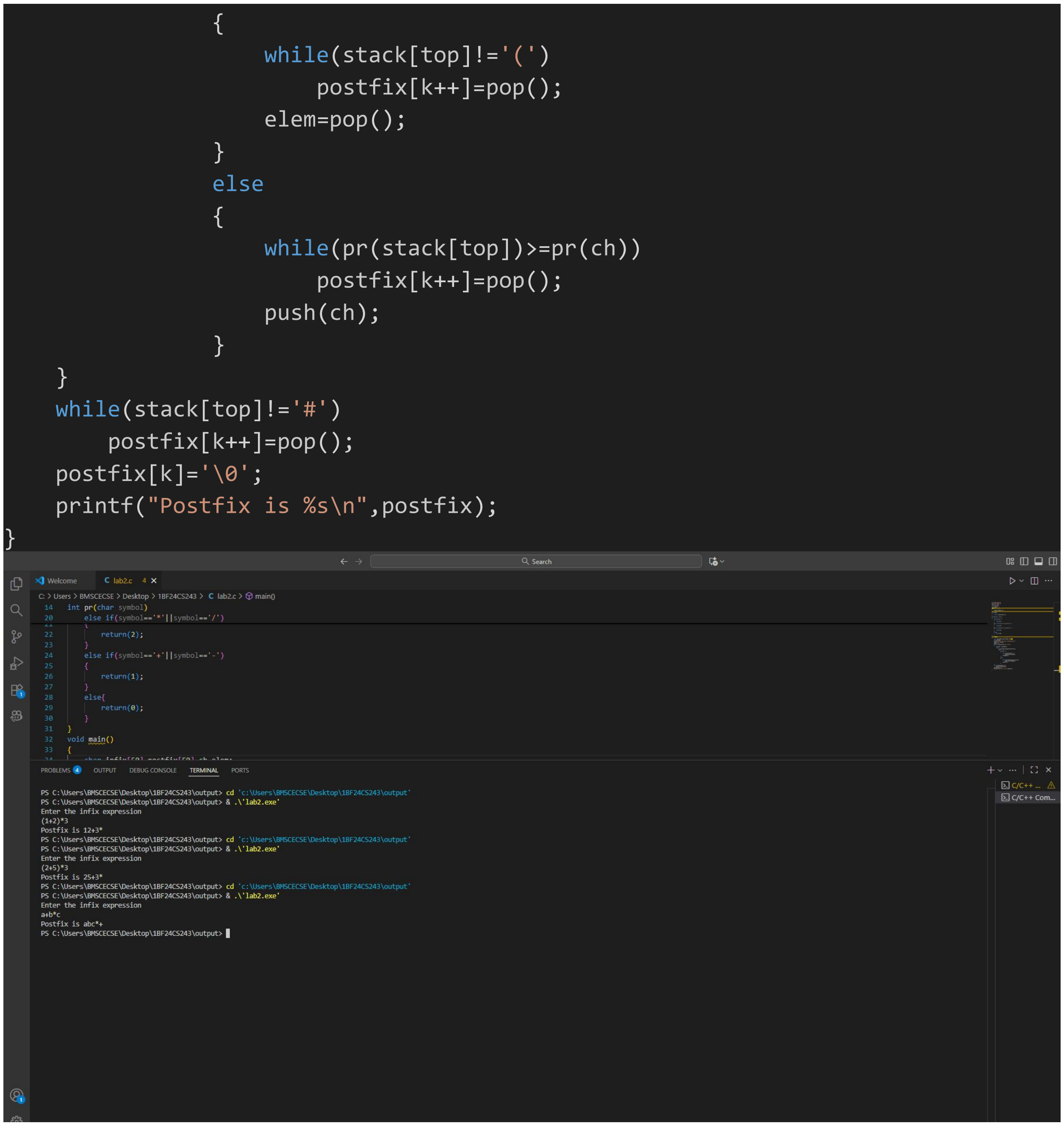
```c
#include<stdio.h>
#include<ctype.h>
#define n 50
char stack[n];
int top=-1;
push(char elem)
{
    stack[++top]=elem;
}
char pop()
{
    return (stack[top--]);
}
int pr(char symbol)
{
    if(symbol=='^')
    {
        return(3);
    }
    else if(symbol=='*'||symbol=='/')
    {
        return(2);
    }
    else if(symbol=='+'||symbol=='-')
    {
        return(1);
    }
    else{
        return(0);
    }
}
void main()
{
    char infix[50],postfix[50],ch,elem;
    int i=0,k=0;
    printf("Enter the infix expression\n");
    scanf("%s",infix);
    push('#');
    while( (ch=infix[i++]) !='\0')
    {
        if(ch=='(')push(ch);
        else
            if(isalnum(ch))postfix[k++]=ch;
            else
                if(ch==')')
```



```c
                {
                    while(stack[top]!='(')
                        postfix[k++]=pop();
                    elem=pop();
                }
                else
                {
                    while(pr(stack[top])>=pr(ch))
                        postfix[k++]=pop();
                    push(ch);
                }
    }
    while(stack[top]!='#')
        postfix[k++]=pop();
    postfix[k]='\0';
    printf("Postfix is %s\n",postfix);
}
```

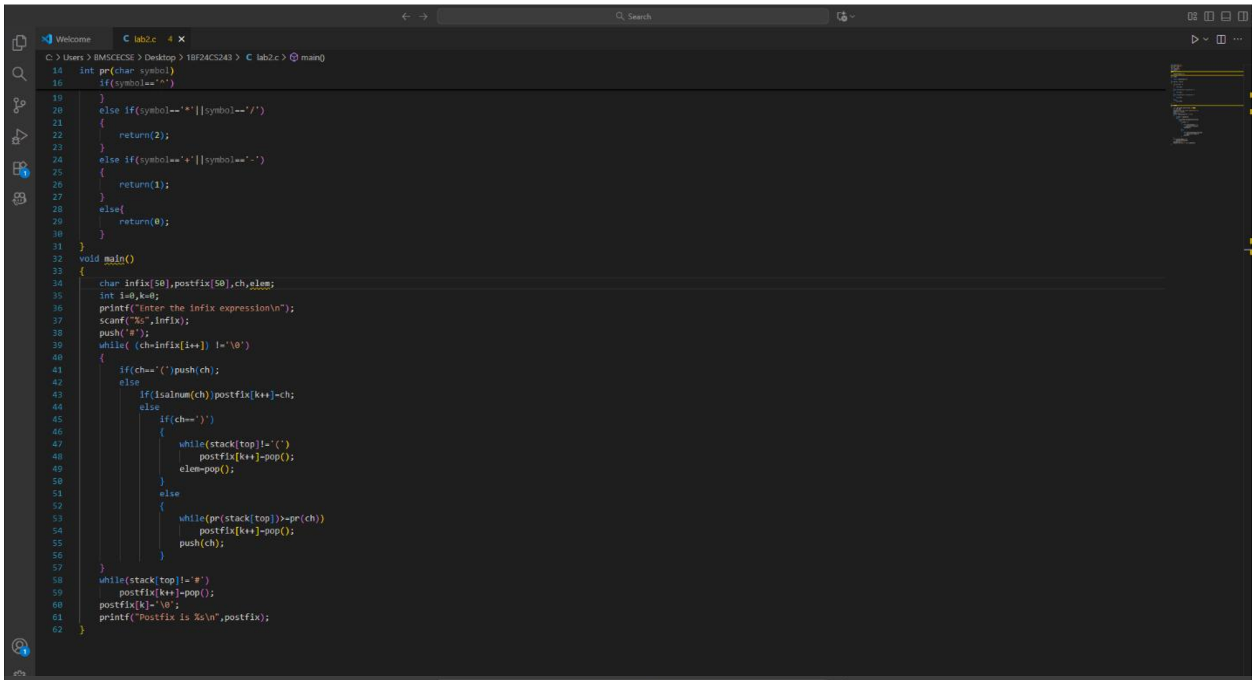
Welcome   lab2.c 4

C: > Users > BMSCECSE > Desktop > 1BF24CS243 > lab2.c > main()

```c
int pr(char symbol)
    else if(symbol=='*'||symbol=='/')
        return(2);
    }
    else if(symbol=='+'||symbol=='-')
    {
        return(1);
    }
    else{
        return(0);
    }
}
void main()
{
```

PROBLEMS 4   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

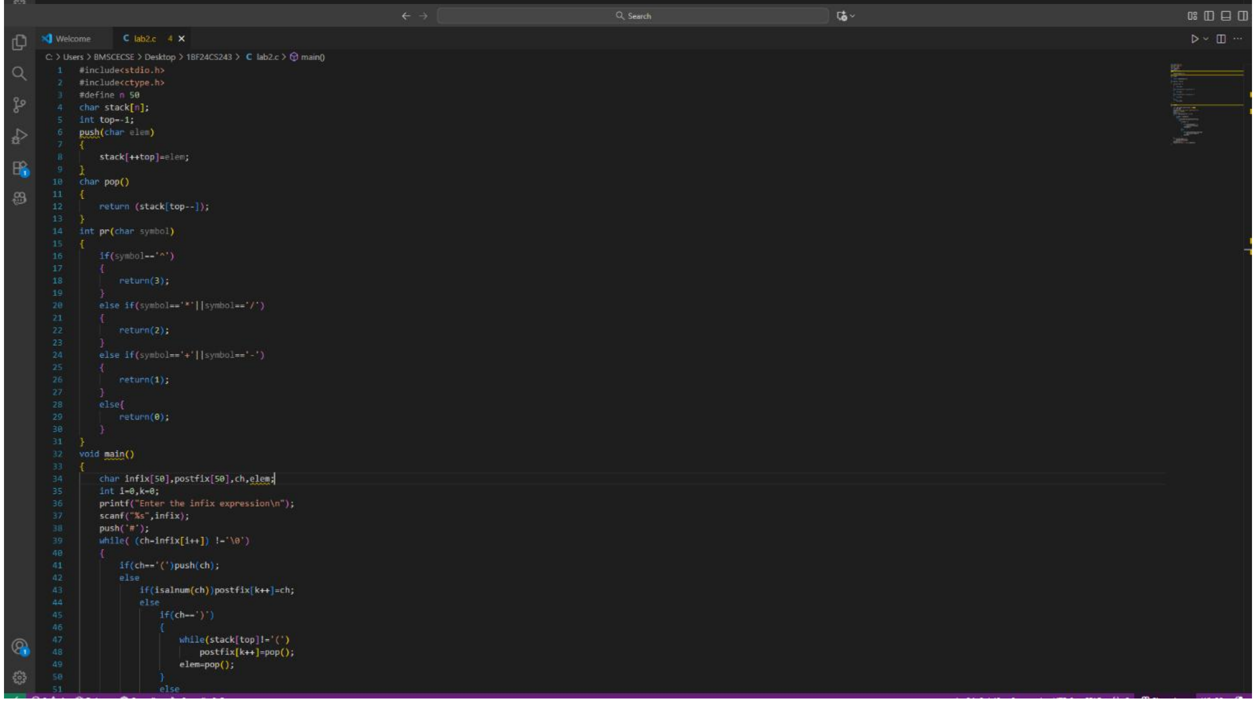
```
PS C:\Users\BMSCECSE\Desktop\1BF24CS243\output> cd 'c:\Users\BMSCECSE\Desktop\1BF24CS243\output'
PS C:\Users\BMSCECSE\Desktop\1BF24CS243\output> & .\'lab2.exe'
Enter the infix expression
(1+2)*3
Postfix is 12+3*
PS C:\Users\BMSCECSE\Desktop\1BF24CS243\output> cd 'c:\Users\BMSCECSE\Desktop\1BF24CS243\output'
PS C:\Users\BMSCECSE\Desktop\1BF24CS243\output> & .\'lab2.exe'
Enter the infix expression
(2+5)*3
Postfix is 25+3*
PS C:\Users\BMSCECSE\Desktop\1BF24CS243\output> cd 'c:\Users\BMSCECSE\Desktop\1BF24CS243\output'
PS C:\Users\BMSCECSE\Desktop\1BF24CS243\output> & .\'lab2.exe'
Enter the infix expression
a+b*c
Postfix is abc*+
PS C:\Users\BMSCECSE\Desktop\1BF24CS243\output> 
```



Welcome | lab2.c 4

C: > Users > BMSCECSE > Desktop > 1BF24CS243 > lab2.c > main()

```c
int pr(char symbol)
    if(symbol=='^')
    }
    else if(symbol=='*'||symbol=='/')
    {
        return(2);
    }
    else if(symbol=='+'||symbol=='-')
    {
        return(1);
    }
    else{
        return(0);
    }
}
void main()
{
    char infix[50],postfix[50],ch,elem;
    int i=0,k=0;
    printf("Enter the infix expression\n");
    scanf("%s",infix);
    push('#');
    while( (ch=infix[i++]) !='\0')
    {
        if(ch=='(')push(ch);
        else
            if(isalnum(ch))postfix[k++]=ch;
            else
                if(ch==')')
                {
                    while(stack[top]!='(')
                        postfix[k++]=pop();
                    elem=pop();
                }
                else
                {
                    while(pr(stack[top])>=pr(ch))
                        postfix[k++]=pop();
                    push(ch);
                }
    }
    while(stack[top]!='#')
        postfix[k++]=pop();
    postfix[k]='\0';
    printf("Postfix is %s\n",postfix);
}
```



Welcome | lab2.c 4

C: > Users > BMSCECSE > Desktop > 1BF24CS243 > lab2.c > main()

```c
#include<stdio.h>
#include<ctype.h>
#define n 50
char stack[n];
int top=-1;
push(char elem)
{
    stack[++top]=elem;
}
char pop()
{
    return (stack[top--]);
}
int pr(char symbol)
{
    if(symbol=='^')
    {
        return(3);
    }
    else if(symbol=='*'||symbol=='/')
    {
        return(2);
    }
    else if(symbol=='+'||symbol=='-')
    {
        return(1);
    }
    else{
        return(0);
    }
}
void main()
{
    char infix[50],postfix[50],ch,elem;
    int i=0,k=0;
    printf("Enter the infix expression\n");
    scanf("%s",infix);
    push('#');
    while( (ch=infix[i++]) !='\0')
    {
        if(ch=='(')push(ch);
        else
            if(isalnum(ch))postfix[k++]=ch;
            else
                if(ch==')')
                {
                    while(stack[top]!='(')
                        postfix[k++]=pop();
                    elem=pop();
                }
                else
```