Ex. No. 3B	EVALUATION OF POSTFIX EXPRESSION
26-07-2017	

Question:

Develop a C++ program to evaluate a postfix expression using stack.

Algorithm:

1. Start.
2. Create a stack array and initialize top=-1
3. Create functions for performing push and pop operations.
4. In main function, create a character array, pointer variable and get postfix expression (token list) from user.
5. Scan the token list from left to right:
   - If the token is an operand, convert it from a string to an integer and push the value onto the Stack.
   - If the token is an operator, *, /, +, or -, it will need two operands. Pop the Stack twice. The first pop is the second operand and the second pop is the first operand. Perform the arithmetic operation. Push the result back on the Stack.
6. When the input expression has been completely processed, the result is on the stack. Pop the Stack and return the value.
7. End.

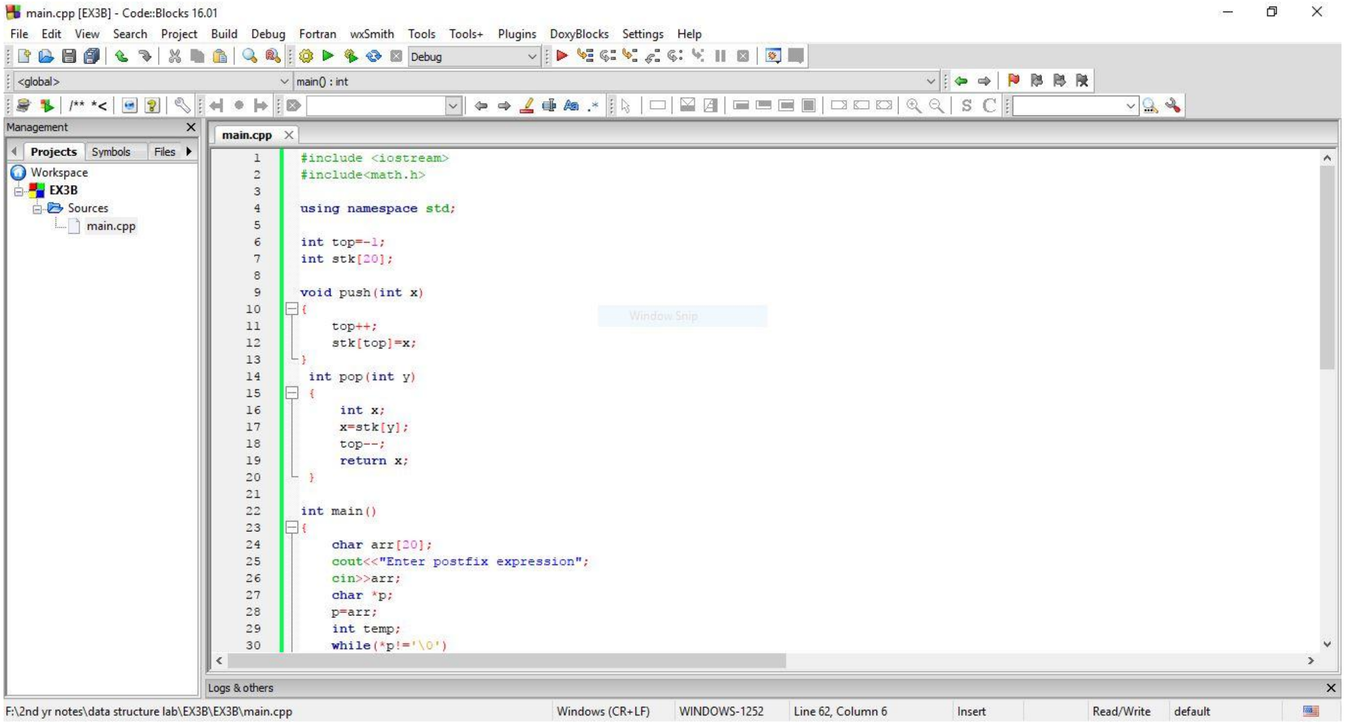
Program:

```
#include <iostream>
#include<math.h>
using namespace std;
int top=-1;
int stk[20];

void push(int x)
{
   top++;
   stk[top]=x;
}
 int pop(int y)
 {
    int x;
    x=stk[y];
    top--;
    return x;
 }
int main()
{  char arr[20];
```

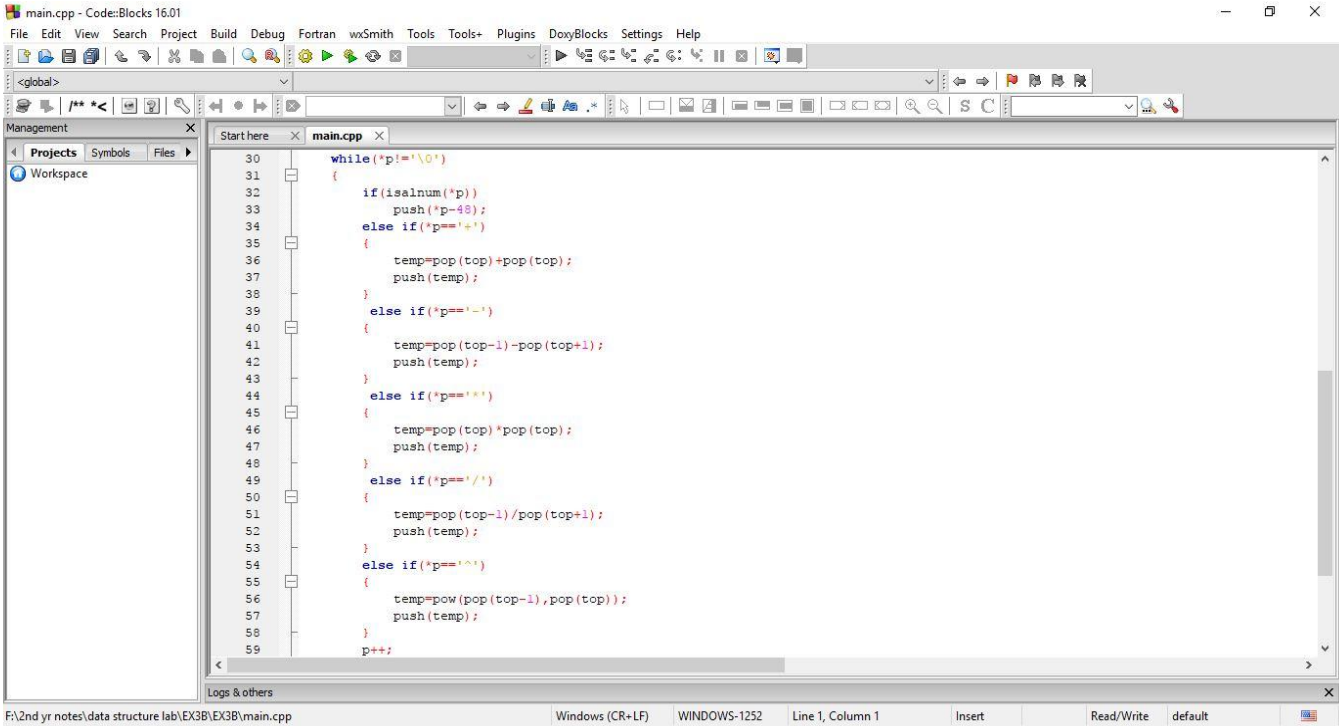
```
cout<<"Enter postfix expression";
cin>>arr;
char *p;
p=arr;
int temp;
while(*p!='\0')
{
    if(isalnum(*p))
        push(*p-48);
    else if(*p=='+')
    {
        temp=pop(top)+pop(top);
        push(temp);
    }
     else if(*p=='-')
    {
        temp=pop(top-1)-pop(top+1);
        push(temp);
    }
     else if(*p=='*')
    {
        temp=pop(top)*pop(top);
```

```
            push(temp);
        }
         else if(*p=='/')
        {
            temp=pop(top-1)/pop(top+1);
            push(temp);
        }
        else if(*p=='^')
        {
            temp=pow(pop(top-1),pop(top));
            push(temp);
        }
        p++;
    }
    while(top!=-1)
    {
        cout<<stk[top];
        top--;
    }
}
```

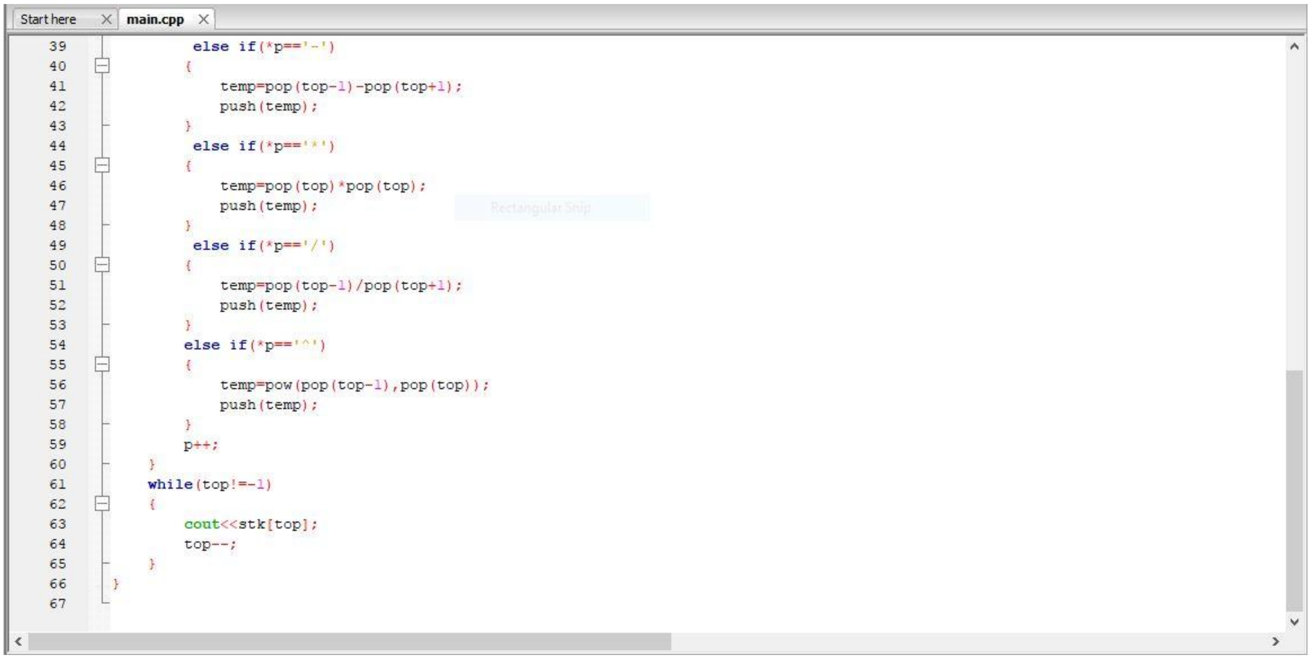
Output:


main.cpp [EX3B] - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Debug
<global>
main() : int
Management
Projects Symbols Files
Workspace
EX3B
Sources
main.cpp
Window Snip
Logs & others
F:\2nd yr notes\data structure lab\EX3B\EX3B\main.cpp
Windows (CR+LF)
WINDOWS-1252
Line 62, Column 6
Insert
Read/Write
default

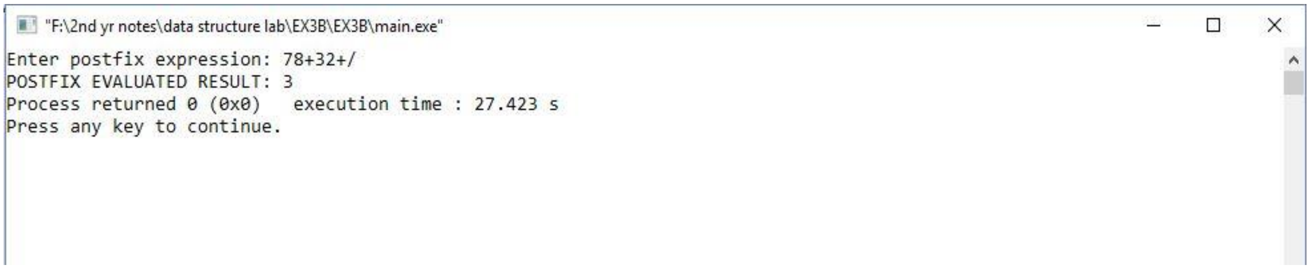
```
#include <iostream>
#include<math.h>

using namespace std;

int top=-1;
int stk[20];

void push(int x)
{
    top++;
    stk[top]=x;
}
 int pop(int y)
 {
     int x;
     x=stk[y];
     top--;
     return x;
 }

int main()
{
    char arr[20];
    cout<<"Enter postfix expression";
    cin>>arr;
    char *p;
    p=arr;
    int temp;
    while(*p!='\0')
```


main.cpp - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
<global>
Management
Projects Symbols Files
Workspace
Start here
main.cpp
Logs & others
F:\2nd yr notes\data structure lab\EX3B\EX3B\main.cpp
Windows (CR+LF)
WINDOWS-1252
Line 1, Column 1
Insert
Read/Write
default

```
    while(*p!='\0')
    {
        if(isalnum(*p))
            push(*p-48);
        else if(*p=='+')
        {
            temp=pop(top)+pop(top);
            push(temp);
        }
         else if(*p=='-')
        {
            temp=pop(top-1)-pop(top+1);
            push(temp);
        }
         else if(*p=='*')
        {
            temp=pop(top)*pop(top);
            push(temp);
        }
         else if(*p=='/')
        {
            temp=pop(top-1)/pop(top+1);
            push(temp);
        }
        else if(*p=='^')
        {
            temp=pow(pop(top-1),pop(top));
            push(temp);
        }
        p++;
```



```
         else if(*p=='-')
        {
            temp=pop(top-1)-pop(top+1);
            push(temp);
        }
         else if(*p=='*')
        {
            temp=pop(top)*pop(top);
            push(temp);
        }
         else if(*p=='/')
        {
            temp=pop(top-1)/pop(top+1);
            push(temp);
        }
        else if(*p=='^')
        {
            temp=pow(pop(top-1),pop(top));
            push(temp);
        }
        p++;
    }
    while(top!=-1)
    {
        cout<<stk[top];
        top--;
    }
}
```



```
Enter postfix expression: 78+32+/
POSTFIX EVALUATED RESULT: 3
Process returned 0 (0x0)   execution time : 27.423 s
Press any key to continue.
```

VIDEO URL:

https://youtu.be/G0A6C5N3dyY

RESULT:

The program of postfix evaluation using stack is implemented successfully and the output is verified.