SIDDHI SINGH

17BIT0028

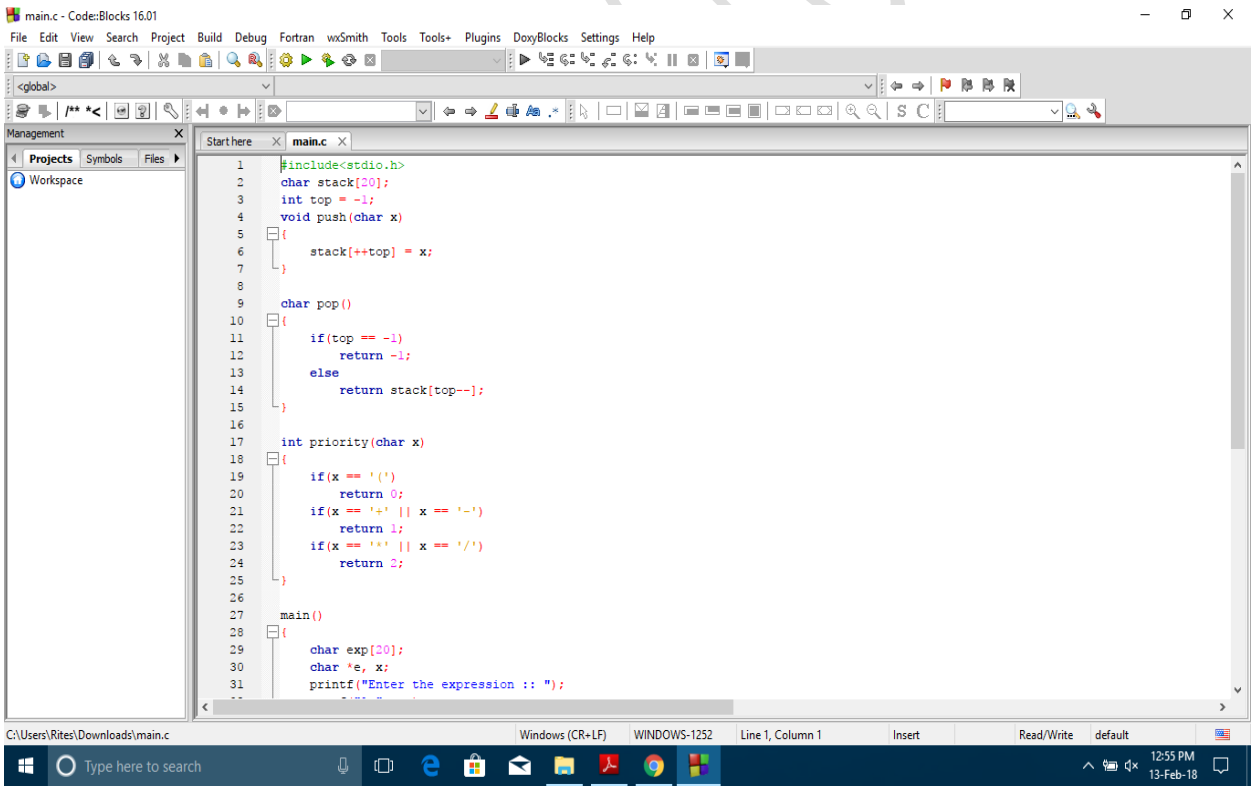
ITE-1004

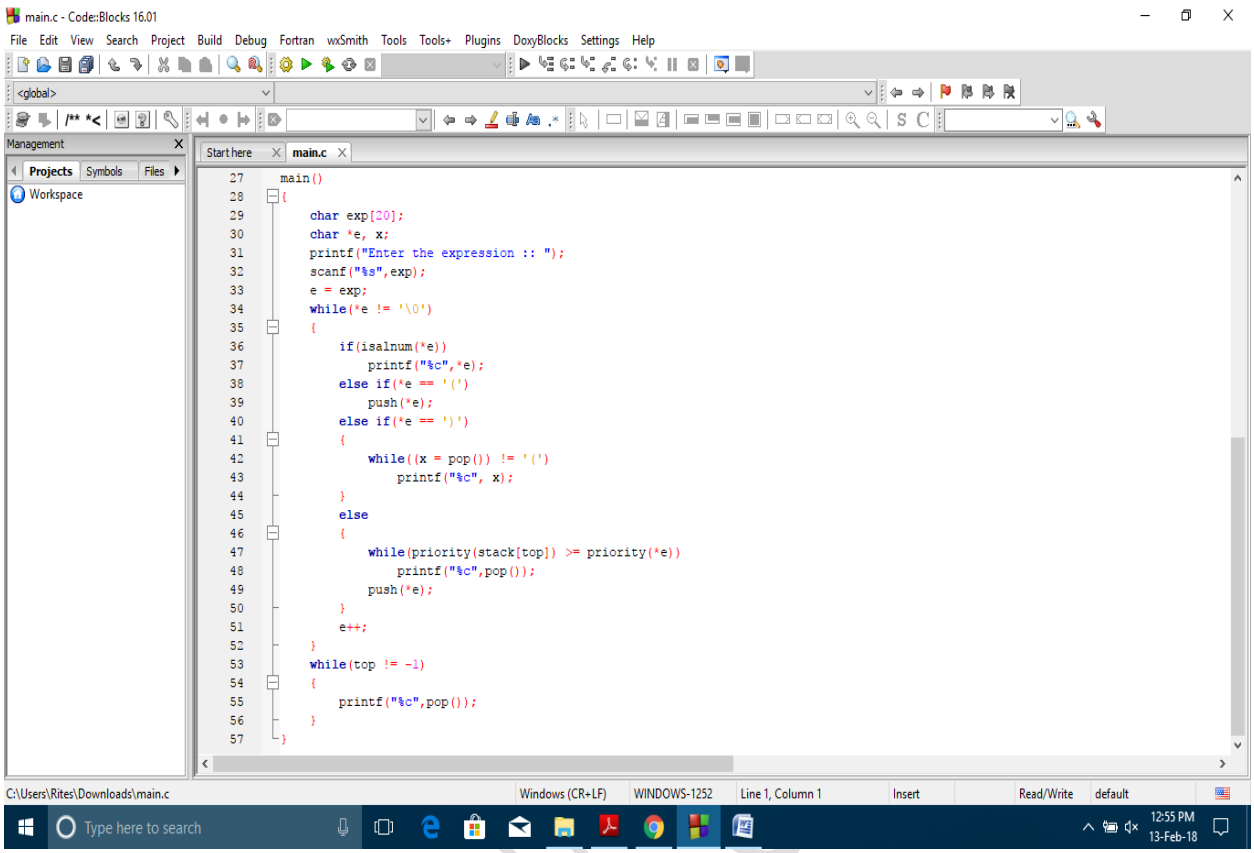
EXERCISE – 4

CODE TO CONVERT INFIX EXPRESSION TO POSTFIX EXPRESSION

```
#include<stdio.h>
char stack[20];
int top = -1;
void push(char x)
{
    stack[++top] = x;
}

char pop()
{
    if(top == -1)
        return -1;
    else
        return stack[top--];
}

int priority(char x)
{
    if(x == '(')
        return 0;
    if(x == '+' || x == '-')
        return 1;
    if(x == '*' || x == '/')
        return 2;
}

main()
{
    char exp[20];
    char *e, x;
    printf("Enter the expression :: ");
    scanf("%s",exp);
    e = exp;
    while(*e != '\0')
    {
        if(isalnum(*e))
            printf("%c",*e);
        else if(*e == '(')
            push(*e);
        else if(*e == ')')
```

```
        {
            while((x = pop()) != '(')
                printf("%c", x);
        }
        else
        {
            while(priority(stack[top]) >= priority(*e))
                printf("%c",pop());
            push(*e);
        }
        e++;
    }
    while(top != -1)
    {
        printf("%c",pop());
    }
}
```





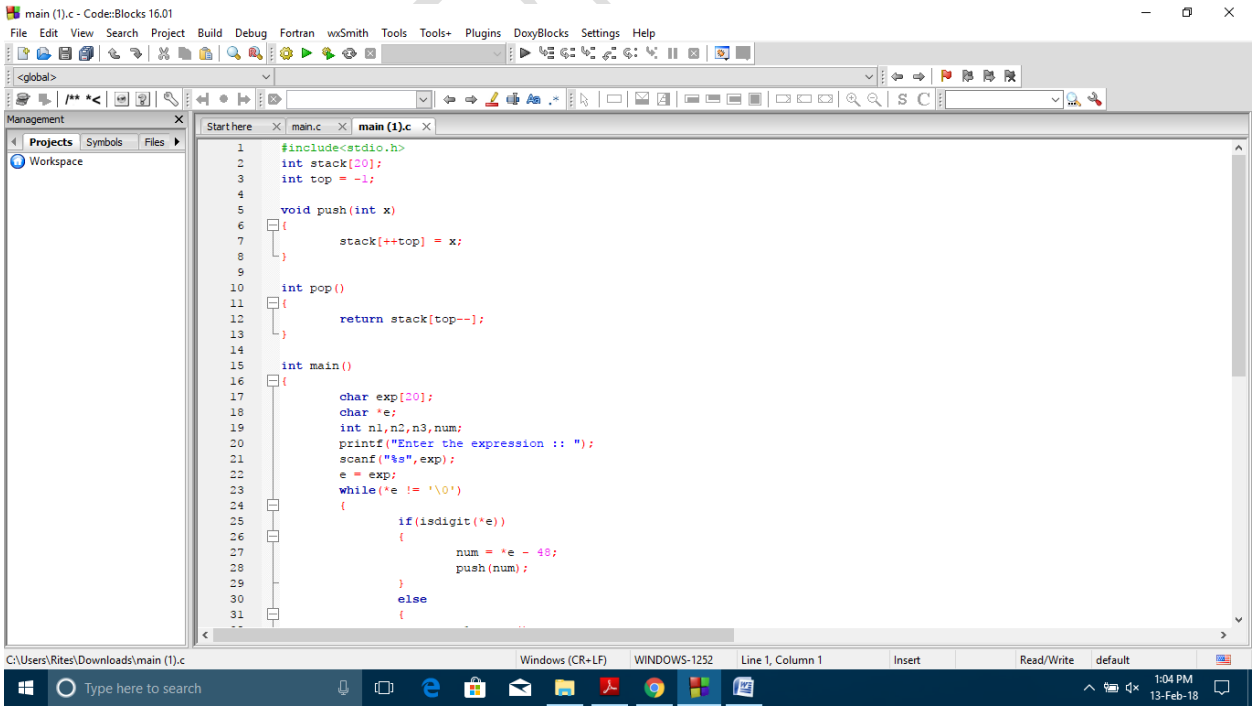
OUTPUT :

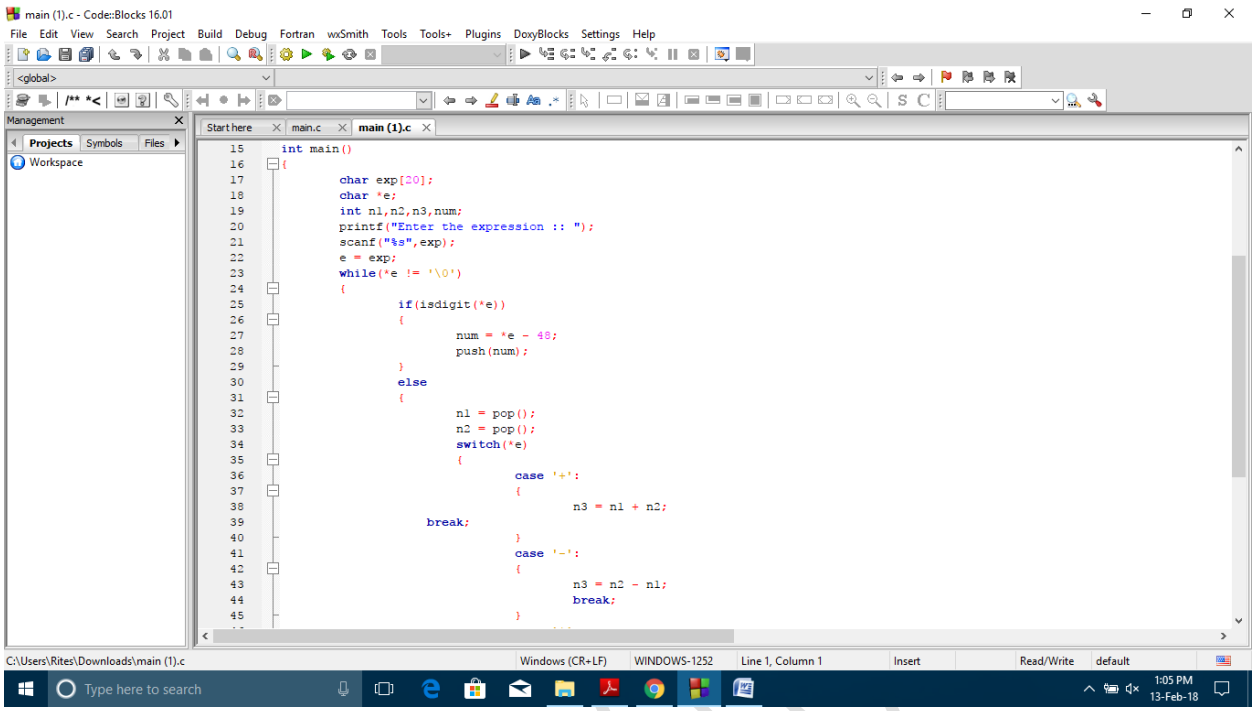
```
gcc version 4.6.3
main.c:27:1: warning: return type defaults to 'int' [-
Wimplicit-int]
 main()
 ^~~~
main.c: In function 'main':
main.c:36:12: warning: implicit declaration of function
'isalnum' [-Wimplicit-function-declaration]
         if(isalnum(*e))
            ^~~~~~~~
Enter the expression ::  (a+b)*(c-d)/(e+f)
ab+cd-*ef+/
```

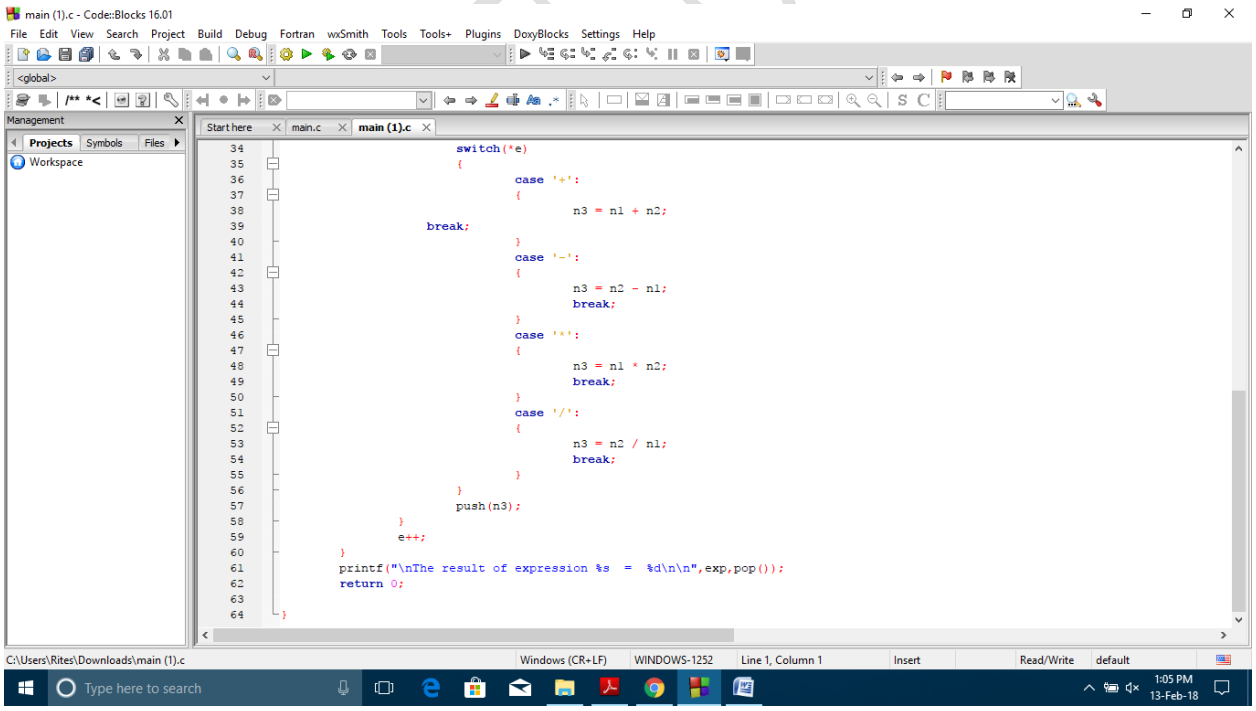
CODE TO EVALUATE POSTFIX EXPRESSION

```
#include<stdio.h>
int stack[20];
int top = -1;

void push(int x)
{
        stack[++top] = x;
}

int pop()
{
        return stack[top--];
}

int main()
{
        char exp[20];
        char *e;
        int n1,n2,n3,num;
        printf("Enter the expression :: ");
        scanf("%s",exp);
        e = exp;
        while(*e != '\0')
        {
                if(isdigit(*e))
                {
                        num = *e - 48;
                        push(num);
                }
                else
                {
                        n1 = pop();
                        n2 = pop();
                        switch(*e)
                        {
                                case '+':
                                {
                                        n3 = n1 + n2;
                                 break;
                                }
                                case '-':
                                {
                                        n3 = n2 - n1;
                                        break;
```

```
                                        }
                                        case '*':
                                        {
                                                n3 = n1 * n2;
                                                break;
                                        }
                                        case '/':
                                        {
                                                n3 = n2 / n1;
                                                break;
                                        }
                                }
                                push(n3);
                        }
                        e++;
                }
                printf("\nThe result of expression %s  =
%d\n\n",exp,pop());
                return 0;

}
```




main (1).c - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Management
Projects Symbols Files
Workspace
Start here
main.c
main (1).c
int main()
{
char exp[20];
char *e;
int n1,n2,n3,num;
printf("Enter the expression :: ");
scanf("%s",exp);
e = exp;
while(*e != '\0')
{
if(isdigit(*e))
{
num = *e - 48;
push(num);
}
else
{
n1 = pop();
n2 = pop();
switch(*e)
{
case '+':
{
n3 = n1 + n2;
break;
}
case '-':
{
n3 = n2 - n1;
break;
}
C:\Users\Rites\Downloads\main (1).c
Windows (CR+LF)
WINDOWS-1252
Line 1, Column 1
Insert
Read/Write
default
Type here to search
1:05 PM
13-Feb-18


main (1).c - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Management
Projects Symbols Files
Workspace
Start here
main.c
main (1).c
switch(*e)
{
case '+':
{
n3 = n1 + n2;
break;
}
case '-':
{
n3 = n2 - n1;
break;
}
case '*':
{
n3 = n1 * n2;
break;
}
case '/':
{
n3 = n2 / n1;
break;
}
}
push(n3);
}
e++;
}
printf("\nThe result of expression %s = %d\n\n",exp,pop());
return 0;
}
C:\Users\Rites\Downloads\main (1).c
Windows (CR+LF)
WINDOWS-1252
Line 1, Column 1
Insert
Read/Write
default
Type here to search
1:05 PM
13-Feb-18

OUTPUT :

```
gcc version 4.6.3
:
main.c: In function 'main':
main.c:25:20: warning: implicit declaration of function
'isdigit' [-Wimplicit-function-declaration]
                if(isdigit(*e))
                   ^~~~~~~~
Enter the expression ::  (300+23)*(43-21)/(84+7)

The result of expression (300+23)*(43-21)/(84+7)  =  12

:
```