Experiment 3

Compiler Design

Aman Kalla

RA1911003010640

**2. a)Write a C program to identify whether a given line is a comment or not.**

#include <stdio.h>

#include <string.h>

#define max 100

void main ()

{

int a,b;

char arr[max];

char dest[2];

printf("Enter the line\n");

scanf("%s",arr);

strncpy(dest,arr,2);

if ((a=strncmp(dest,"//",2)) && (b=strncmp(dest,"/\*",2)))

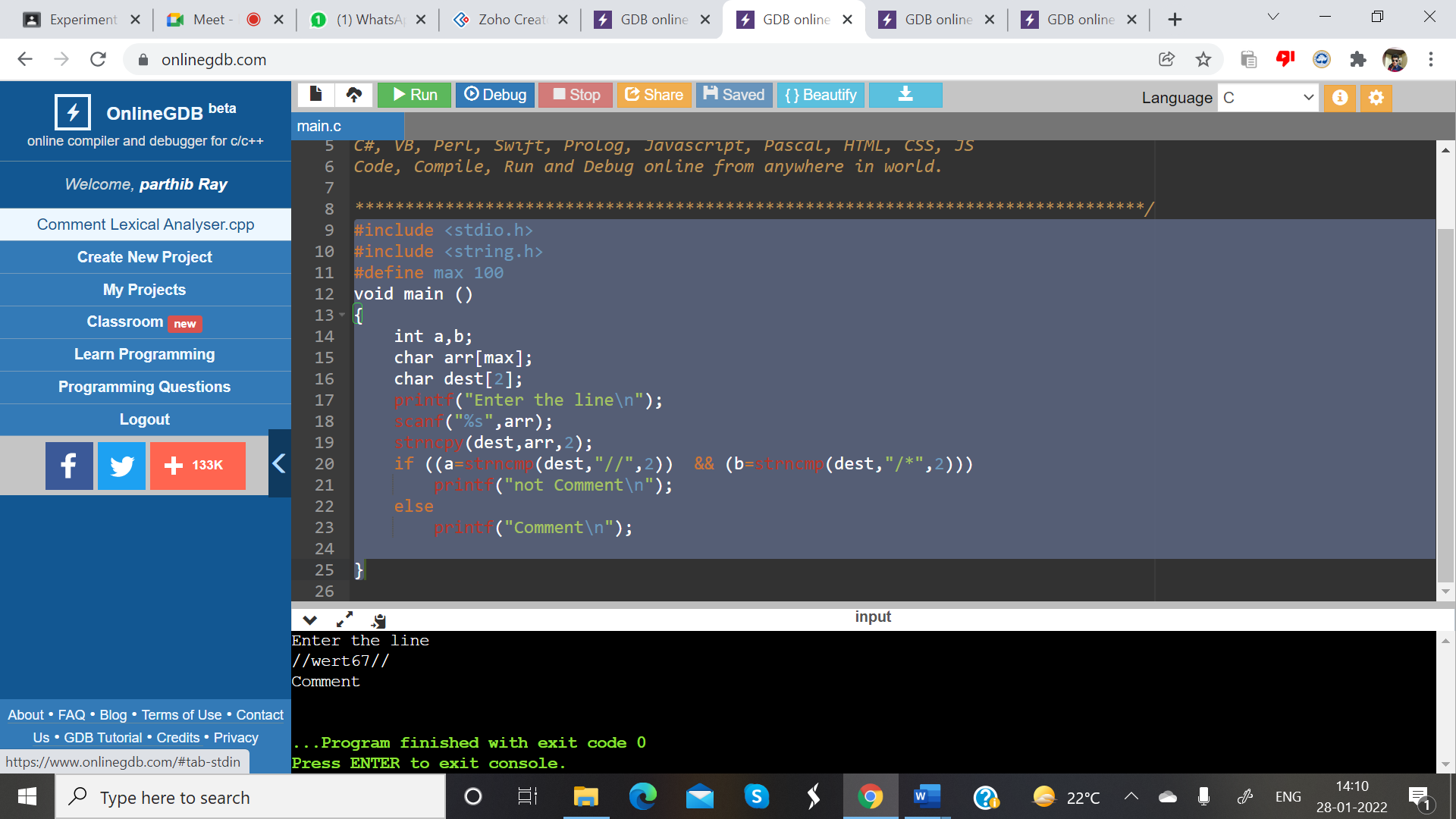
printf("not Comment\n");

else

printf("Comment\n");

}

Input and Output:-



**2 b)  b)Write a C program to simulate lexical analyzer for validating operators.**

#include<stdio.h>

#include<conio.h>

void main()

{

char s[5];

printf("\n Enter any operator:");

scanf("%s",s);

switch(s[0])

{

case'>': if(s[1]=='=')

printf("\n Greater than or equal");

else

printf("\n Greater than");

break;

case'<': if(s[1]=='=')

printf("\n Less than or equal");

else

printf("\nLess than");

break;

case'=': if(s[1]=='=')

printf("\nEqual to");

else

printf("\nAssignment");

break;

case'!': if(s[1]=='=')

printf("\nNot Equal");

else

printf("\n Bit Not");

break;

case'&': if(s[1]=='&')

printf("\nLogical AND");

else

printf("\n Bitwise AND");

break;

case'|': if(s[1]=='|')

printf("\nLogical OR");

else

printf("\nBitwise OR");

break;

case'+': printf("\n Addition");

break;

case'-': printf("\nSubstraction");

break;

case'\*': printf("\nMultiplication");

break;

case'/': printf("\nDivision");

break;

case'%': printf("Modulus");

break;

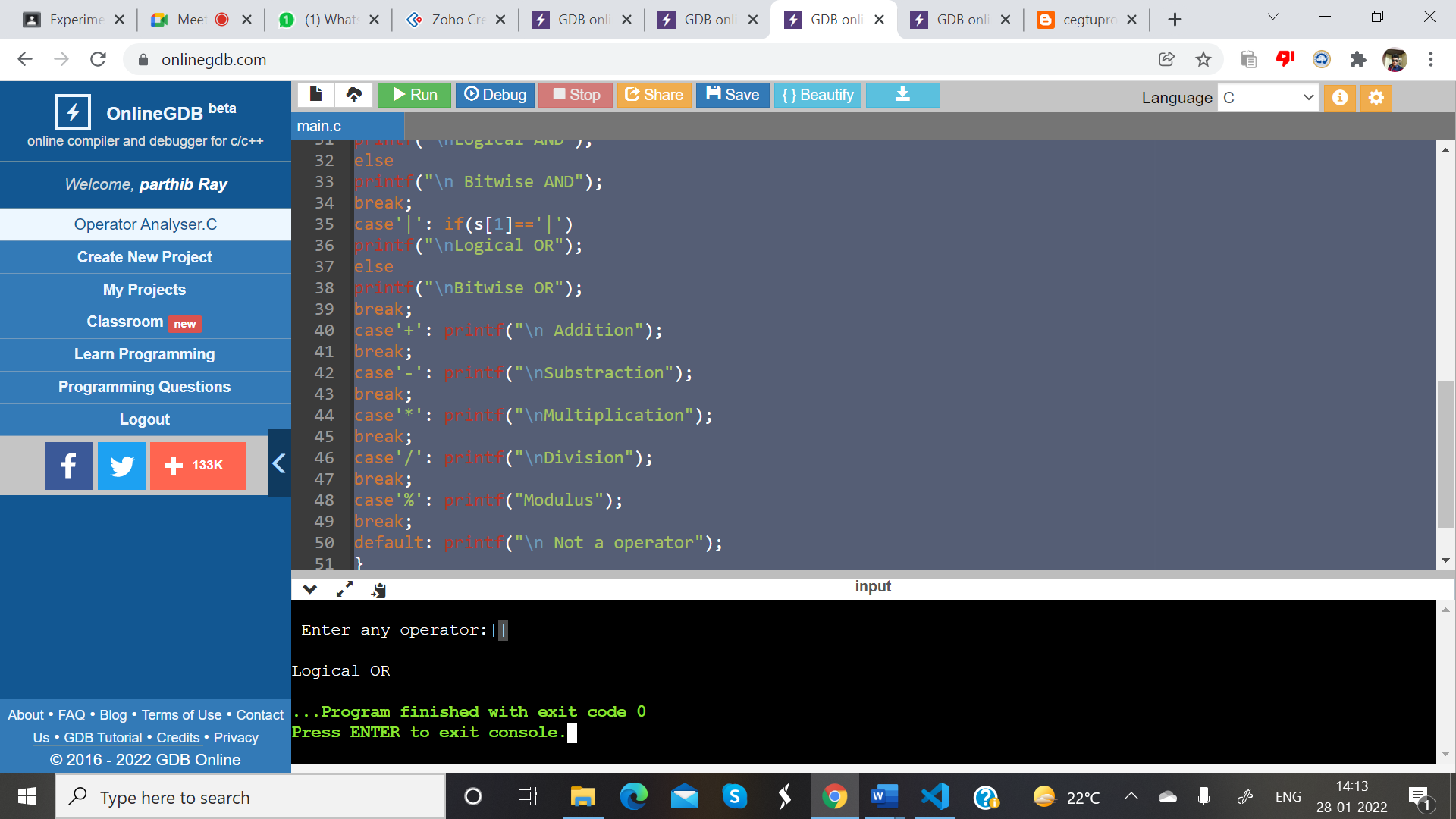
default: printf("\n Not a operator");

}

getch();

}

**Input and Output:-**



**2 c)** **Write a C program to recognize strings under 'a', 'a\*b+', 'abb'.**

#include<stdio.h>

#include<conio.h>

#include<string.h>

#include<stdlib.h>

void main()

{

char s[20],c;

int state=0,i=0;

printf("\n Enter a string:");

scanf("%s",s);

while(s[i]!='\0')

{

switch(state)

{

case 0: c=s[i++];

if(c=='a')

state=1;

else if(c=='b')

state=2;

else

state=6;

break;

case 1: c=s[i++];

if(c=='a')

state=3;

else if(c=='b')

state=4;

else

state=6;

break;

case 2: c=s[i++];

if(c=='a')

state=6;

else if(c=='b')

state=2;

else

state=6;

break;

case 3: c=s[i++];

if(c=='a')

state=3;

else if(c=='b')

state=2;

else

state=6;

break;

case 4: c=s[i++];

if(c=='a')

state=6;

else if(c=='b')

state=5;

else

state=6;

break;

case 5: c=s[i++];

if(c=='a')

state=6;

else if(c=='b')

state=2;

else

state=6;

break;

case 6: printf("\n %s is not recognised.",s);

exit(0);

}

}

if(state==1)

printf("\n %s is accepted under rule 'a'",s);

else if((state==2)||(state==4))

printf("\n %s is accepted under rule 'a\*b+'",s);

else if(state==5)

printf("\n %s is accepted under rule 'abb'",s);

getch();

}

**Input and Output:-**

