Delhi Skill and Entrepreneurship University

**YEAR: - 2021-2022**



**Ambedkar institute of technology**

**shakarapur**

BCA-B

**PROBLEM SOLVING PROGRAMMING IN C**

**Faculty Name- Mrs. Avinash kaur**

**Name- Vidhi Verma**

**Roll no.- 41221184**

Question 1. WAP to find the ascii value of a character.

Code: -

#include <stdio.h>

int

main ()

{

char ch;

printf ("Enter a character");

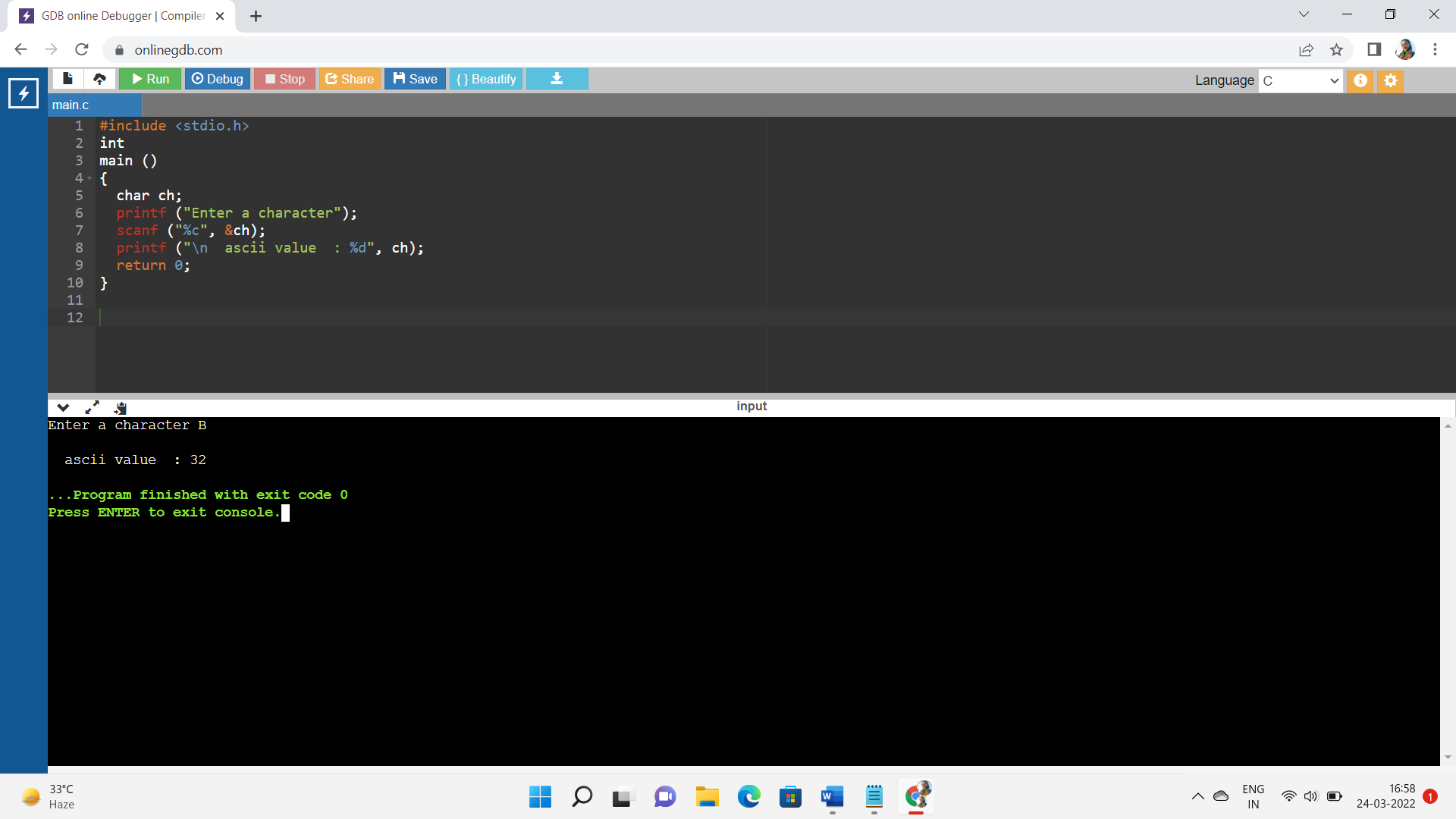
scanf ("%c", &ch);

printf ("\n ascii value : %d", ch);

return 0;

}

Output: -



Question 2. WAP to find sum of digits.

Code: -

#include <stdio.h>

int main()

{

int a,b,c,sum;

printf("Enter Two Digit Number: ");

scanf("%d",&a);

b=a%10;

c=a/10;

printf("\nFirst Digit = %d \nLast Digit = %d\n",c,b);

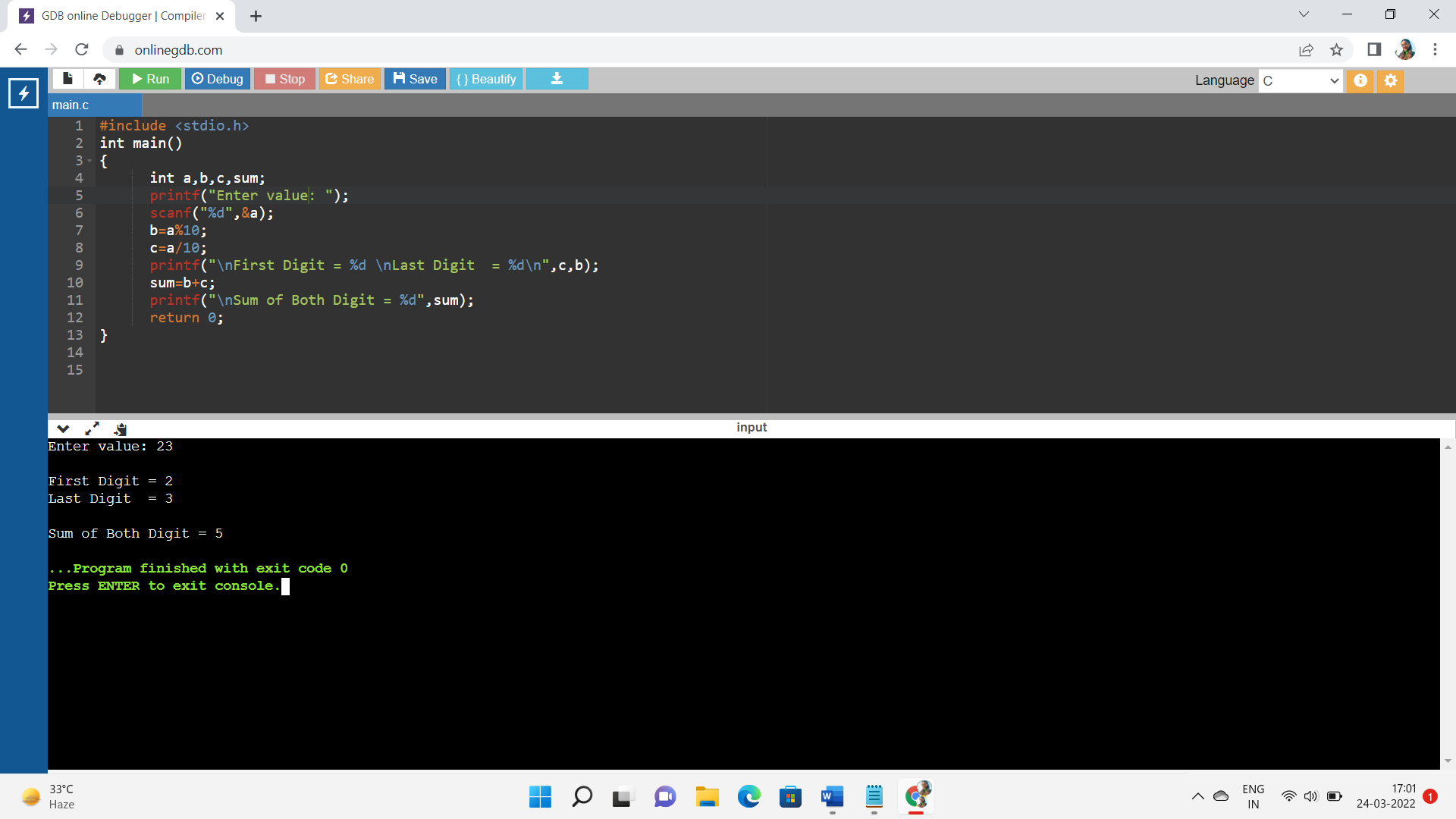
sum=b+c;

printf("\nSum of Both Digit = %d",sum);

return 0;

}

Output: -



Question 3:- WAP to find largest number among n numbers.

Code:-

#include <stdio.h>

int main()

{

int a,b,c;

printf("enter value");

scanf("%d", &a);

printf("enter value");

scanf("%d", &b);

printf("enter value");

scanf("%d", &c);

if(a>b&&a>c)

printf("%d is the largest number", a);

else if(b>a&&b>c)

printf("%d is the largest number", b);

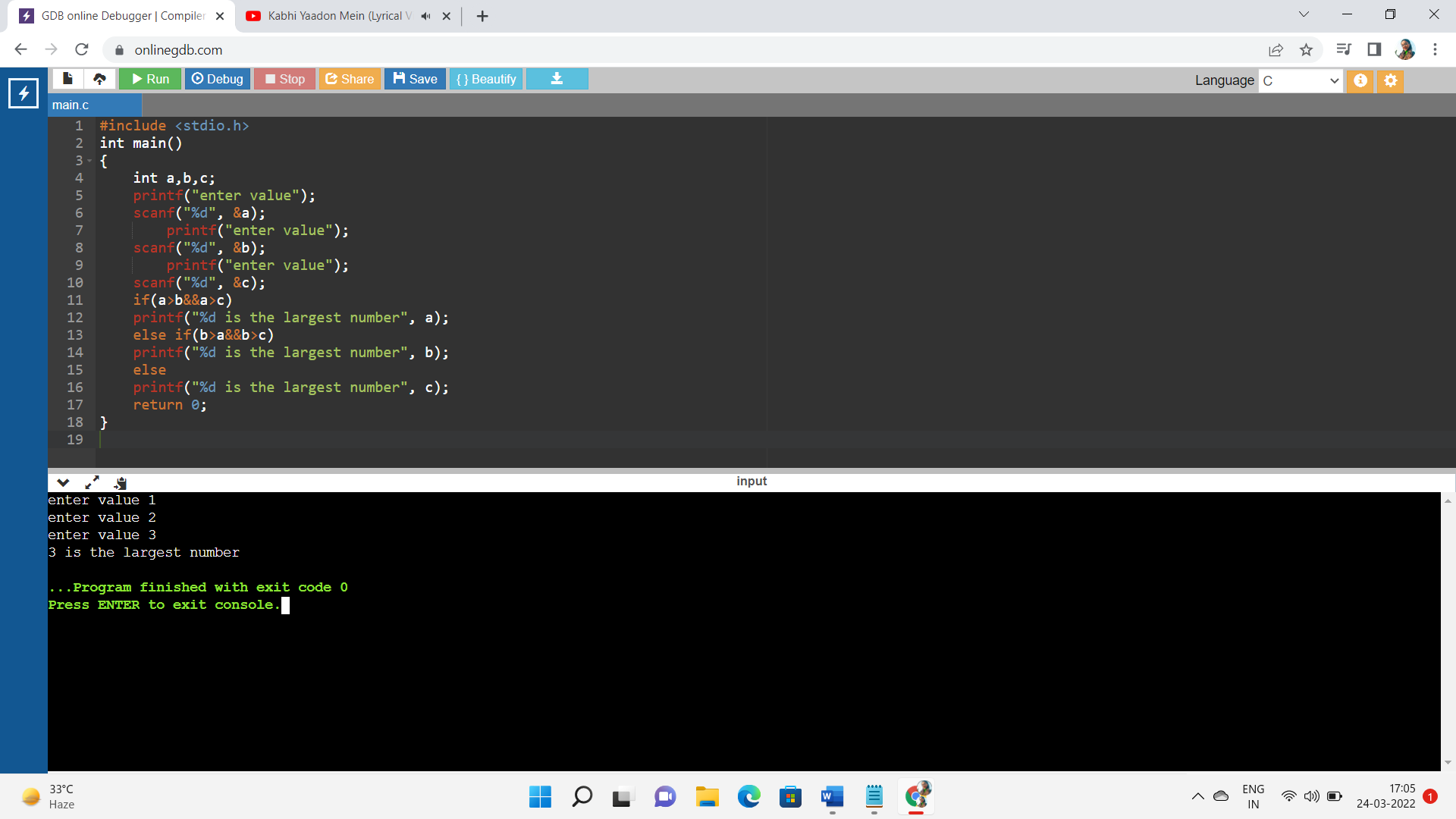
else

printf("%d is the largest number", c);

return 0;

}

Output:-



Question 4. WAP to Insert an element in an Array.

Code:-

#include <stdio.h>

int main()

{ int arr[30] = { 0 };

int i, x, pos, n = 5;

for (i = 0; i < 5; i++)

arr[i] = i + 1;

x = 7;

pos = 2;

n++;

for (i = n-1; i >= pos; i--)

arr[i] = arr[i - 1];

arr[pos - 1] = x;

for (i = 0; i < n; i++)

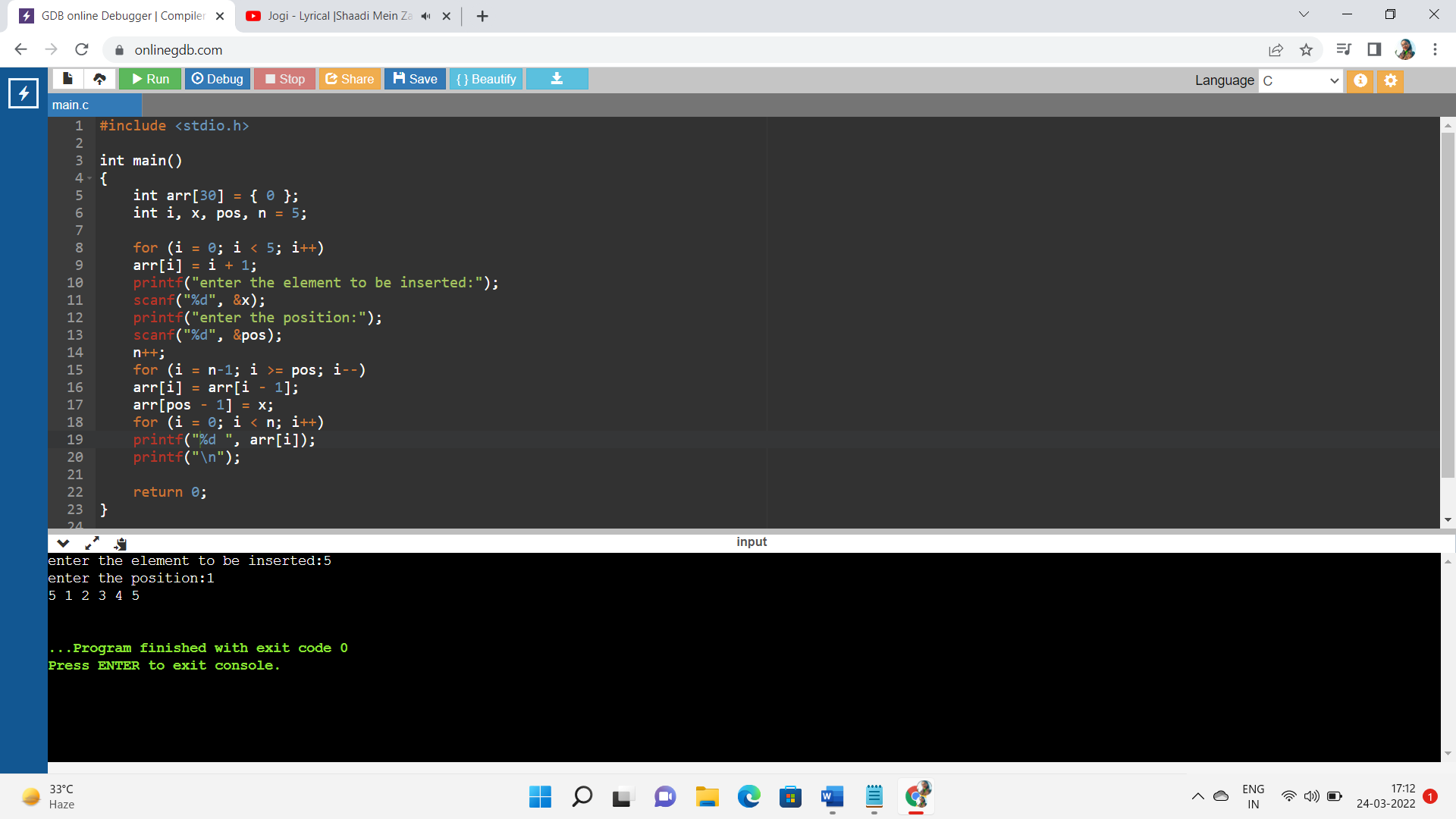
printf("%d ", arr[i]);

printf("\n");

return 0;

}

Output:-



Question 5. WAP to find the largest and smallest element in a array.

Code:- #include <stdio.h>

int main()

{ int arr[100], n, i, small, large;

printf("Enter the number of elements : ");

scanf("%d", &n);

for (i = 0; i < n; i++)

{

printf("Enter element %d : ", i + 1);

scanf("%d", &arr[i]);

}

small = arr[0];

large = arr[0];

for (i = 1; i < n; i++)

{

if (arr[i] < small)

small = arr[i];

if (arr[i] > large)

large = arr[i];

}

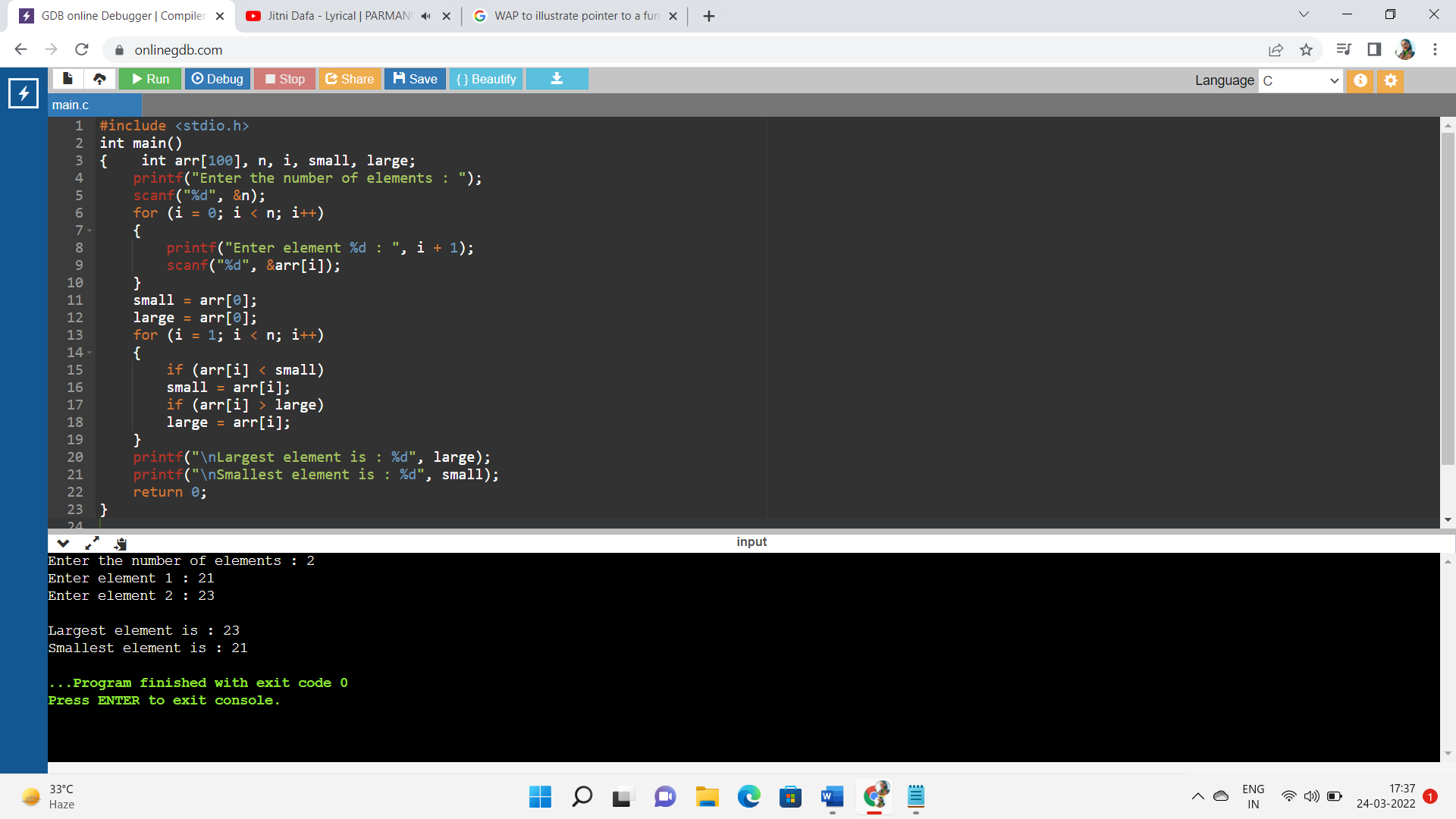
printf("\nLargest element is : %d", large);

printf("\nSmallest element is : %d", small);

return 0;

}

Output:-



Question 6. WAP to6 find the reverse of string.

Code:- #include <stdio.h>

#include <string.h>

void revstr(char \*str1)

{

int i, len, temp;

len = strlen(str1);

for (i = 0; i < len/2; i++)

{

temp = str1[i];

str1[i] = str1[len - i - 1];

str1[len - i - 1] = temp;

}

} int main()

{

char str[50];

printf (" Enter the string: ");

gets(str);

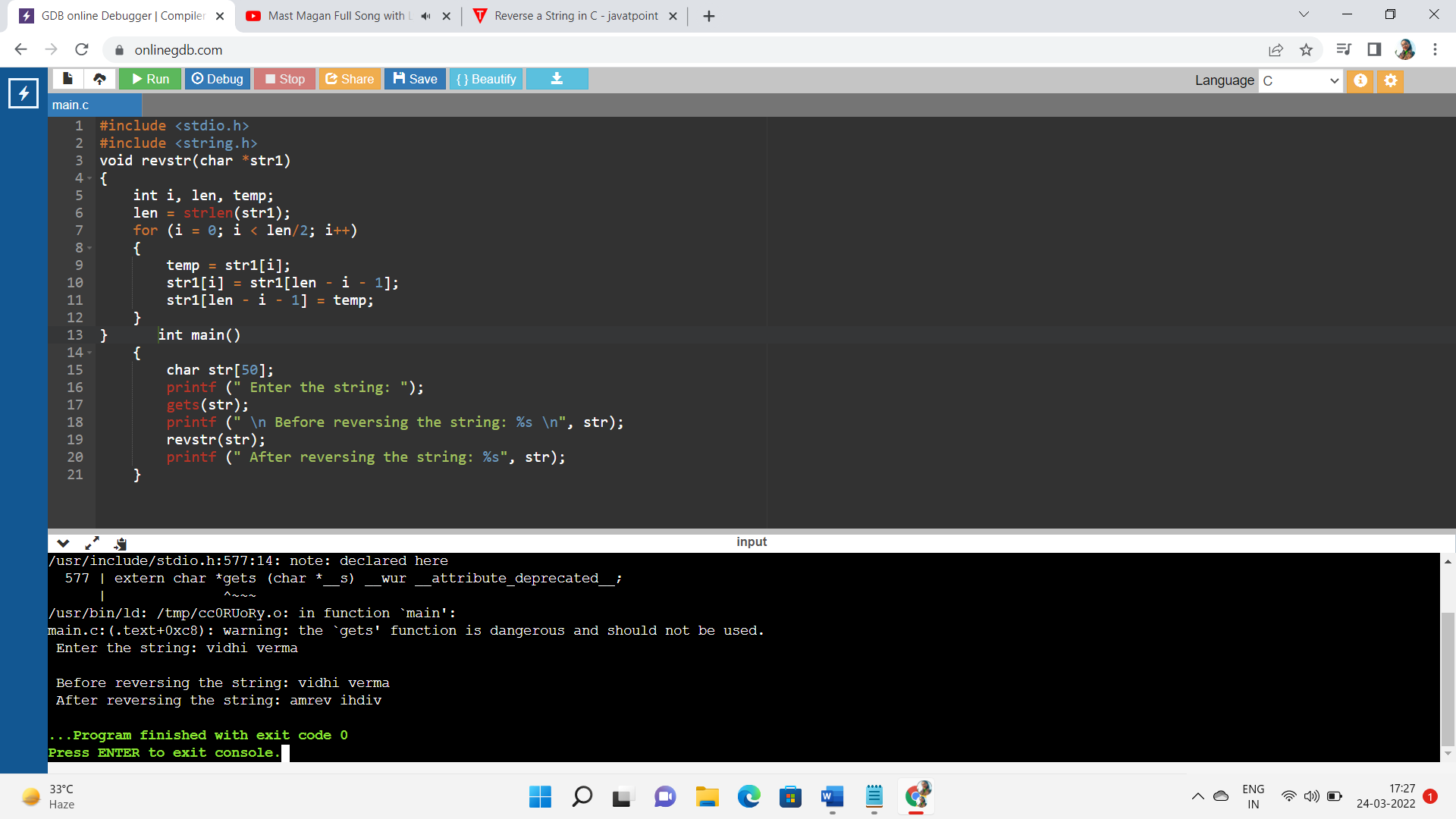
printf (" \n Before reversing the string: %s \n", str);

revstr(str);

printf (" After reversing the string: %s", str);

}

Output:-



Question 7. WAP to illustrate pointer to a function.

Code:-

#include <stdio.h>

int\* larger(int\*, int\*);

void main()

{

int a = 15;

int b = 92;

int \*p;

p = larger(&a, &b);

printf("%d is larger",\*p);

}

int\* larger(int \*x, int \*y)

{

if(\*x > \*y)

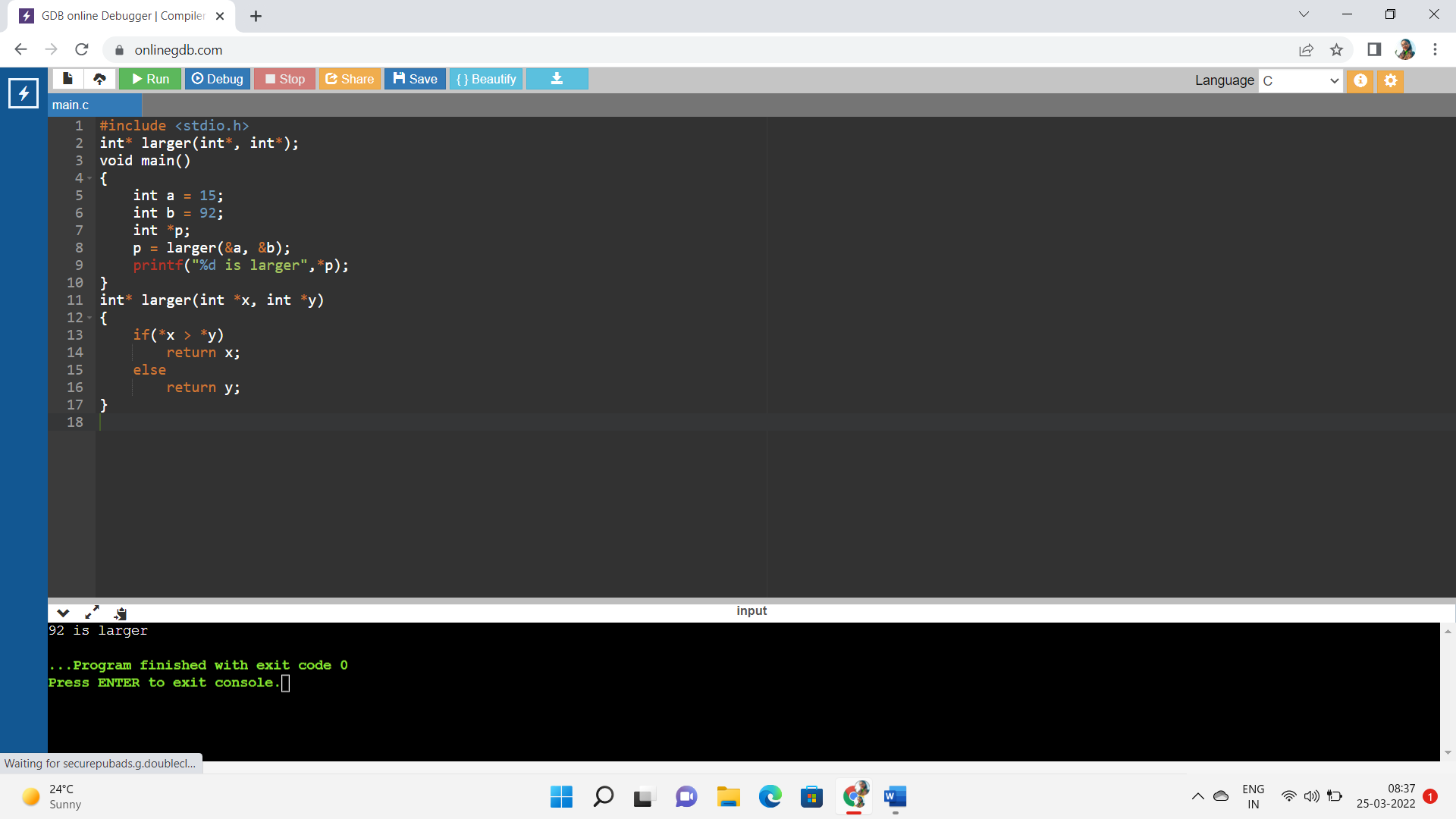
return x;

else

return y;

}

Output: -



Question 8. WAP to demonstrate array of a pointer.

Code: -#include <stdio.h>

int main()

{ int a,b,c;

int \*ptr[3];

ptr[0]= &a;

ptr[1]= &b;

ptr[2]= &c;

a=100;

b=200;

c=300;

printf("value of a: %d, b: %d, c: %d\n",\*ptr[0],\*ptr[1],\*ptr[2]);

\*ptr[0] +=10;

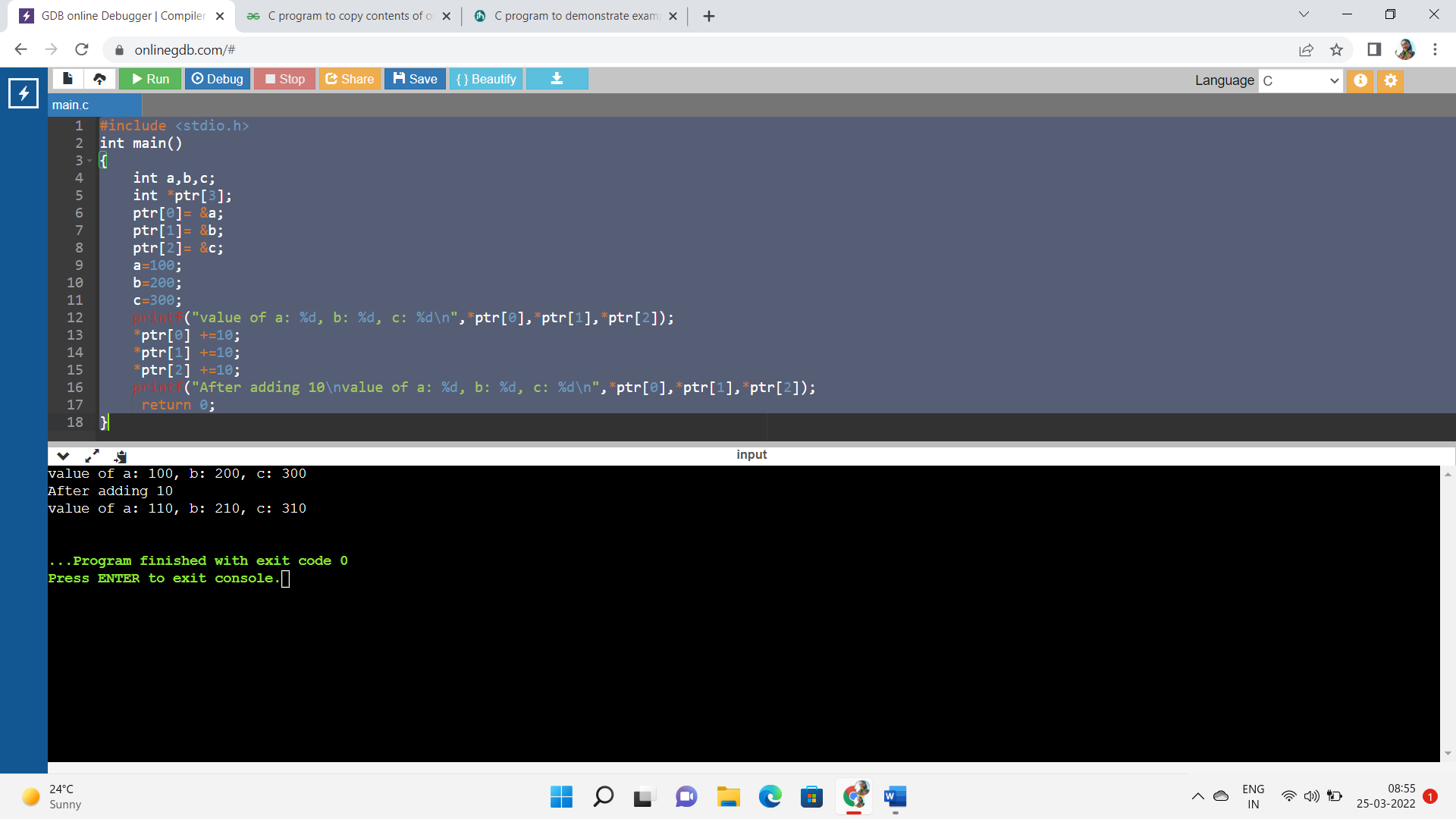
\*ptr[1] +=10;

\*ptr[2] +=10;

printf("After adding 10\nvalue of a: %d, b: %d, c: %d\n",\*ptr[0],\*ptr[1],\*ptr[2]);

return 0;}

Output: -



Question 9. WAP to copy file to another file.

Code: -

#include <stdio.h>

#include <stdlib.h>

int main()

{

FILE \*fptr1, \*fptr2;

char filename[100], c;

printf("Enter the filename to open for reading \n");

scanf("%s", filename);

fptr1 = fopen(filename, "r");

if (fptr1 == NULL)

{

printf("Cannot open file %s \n", filename);

exit(0);

}

printf("Enter the filename to open for writing \n");

scanf("%s", filename);

fptr2 = fopen(filename, "w");

if (fptr2 == NULL)

{

printf("Cannot open file %s \n", filename);

exit(0);

}

c = fgetc(fptr1);

while (c != EOF)

{

fputc(c, fptr2);

c = fgetc(fptr1);

}

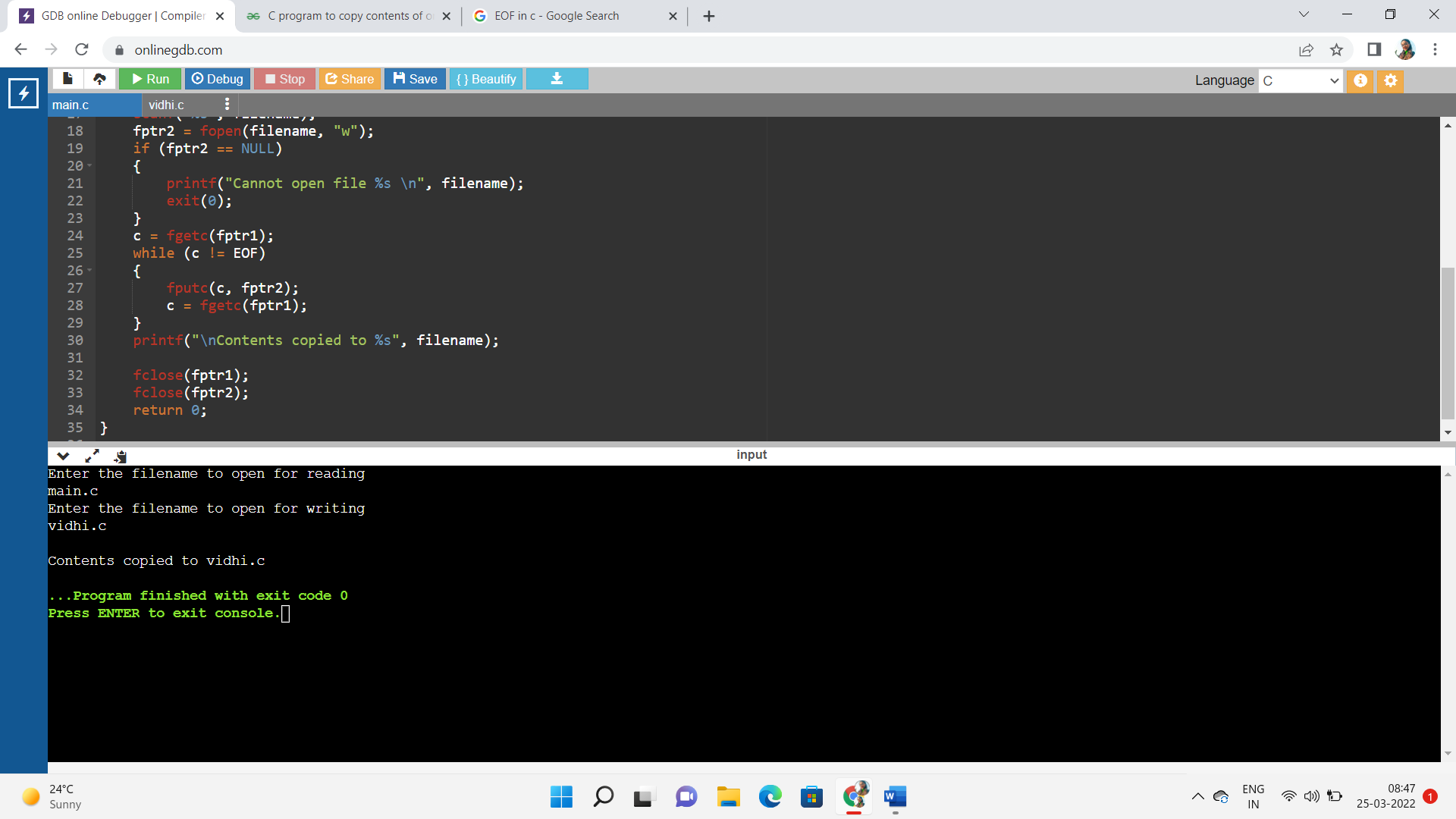
printf("\nContents copied to %s", filename);

fclose(fptr1);

fclose(fptr2);

return 0;

}

Output: -

Question 10. WAP to create a structure of teacher data (minimum four members);

Code: -

#include <stdio.h>

struct student {

char name[50];

int salary;

int contact;

int ID;

} s;

int main() {

printf("Enter information:\n");

printf("Enter name: ");

fgets(s.name, sizeof(s.name), stdin);

printf("Enter ID: ");

scanf("%d", &s.ID);

printf("Enter salary: ");

scanf("%d", &s.salary);

printf("enter contact number:");

scanf("%d", &s.contact);

printf("Displaying Information:\n");

printf("Name: ");

printf("%s", s.name);

printf("ID: %d\n", s.ID);

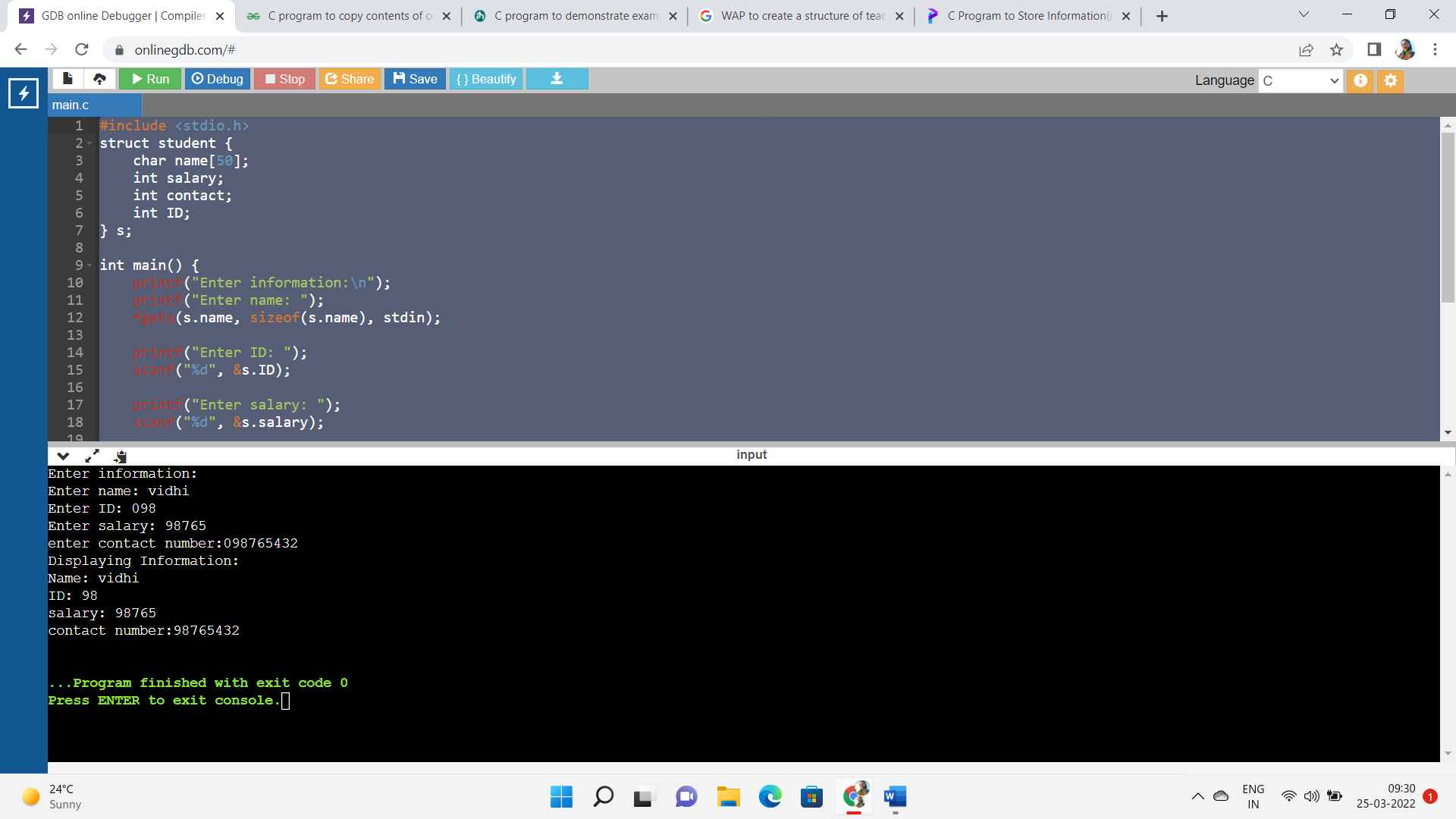
printf("salary: %d\n", s.salary);

printf("contact number:%d\n", s.contact);

return 0;

}

Output: -



Question 11. WAP to demonstrate the difference between union and structure.

Code: -

#include <stdio.h>

struct Employee

{

int age;

char Name[50];

char Department[20];

float Salary;

} emp1;

union Person

{

int ag;

char Nam[50];

char Departent[20];

float Salar;

} Person1;

int main()

{

struct Employee emp1;

union Person Person1;

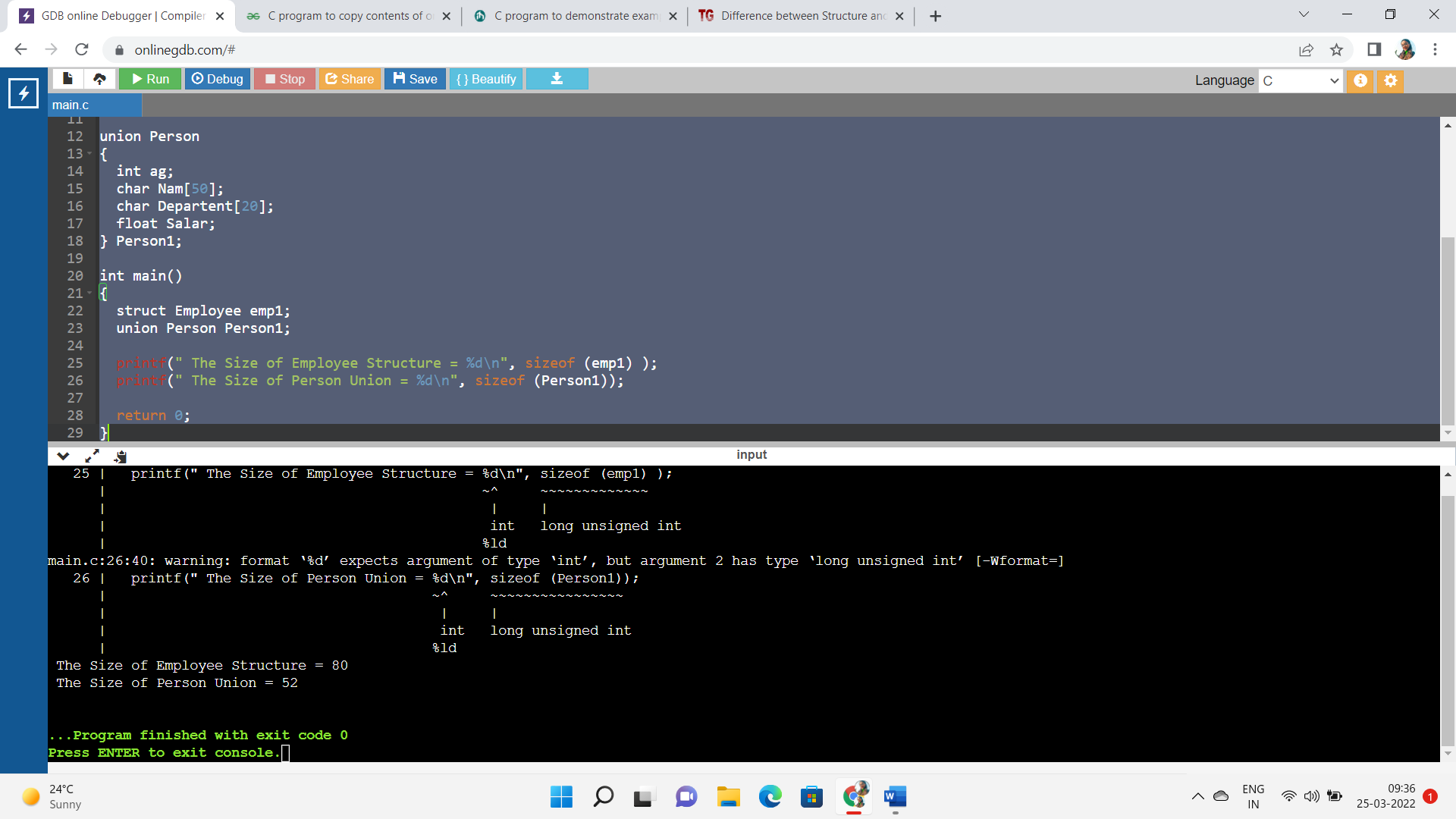
printf(" The Size of Employee Structure = %d\n", sizeof (emp1) );

printf(" The Size of Person Union = %d\n", sizeof (Person1));

return 0;

}

Output: -



Question 12. WAP to demonstrate DMA.

Code: -

#include <stdio.h>

#include <stdlib.h>

int main() {

int n, i, \*ptr, sum = 0;

printf("Enter number of elements: ");

scanf("%d", &n);

ptr = (int\*) malloc(n \* sizeof(int));

if(ptr == NULL) {

printf("Error! memory not allocated.");

exit(0);

}

printf("Enter elements: ");

for(i = 0; i < n; ++i) {

scanf("%d", ptr + i);

sum += \*(ptr + i);

}

printf("Sum = %d", sum);

free(ptr);

return 0;

}

Output: -

