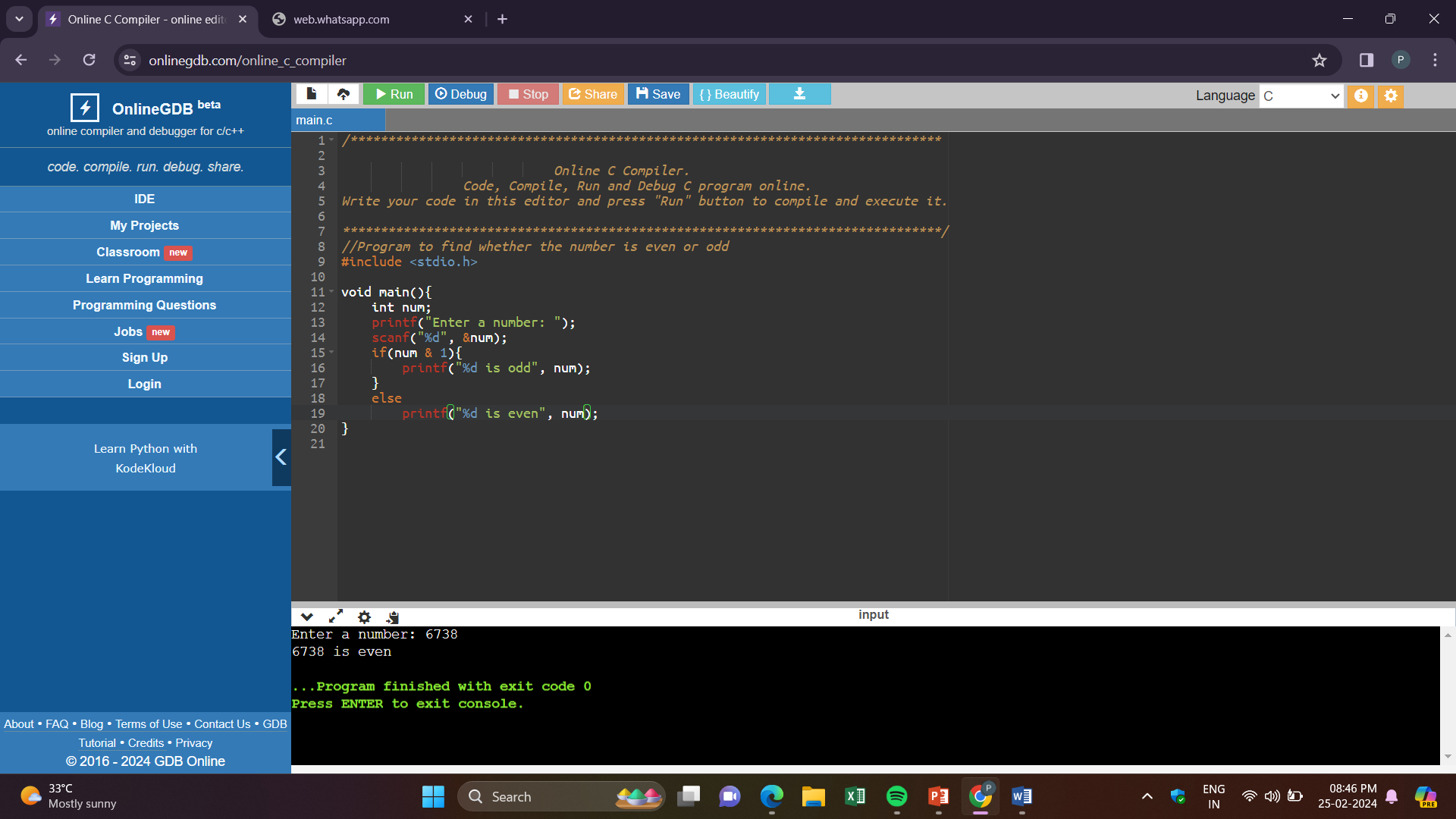
ADVANCED C PROGRAMMING – MODULE 1 ASSESSMENT

1. Write a C program to determine the given number is odd or even using Bitwise operators.



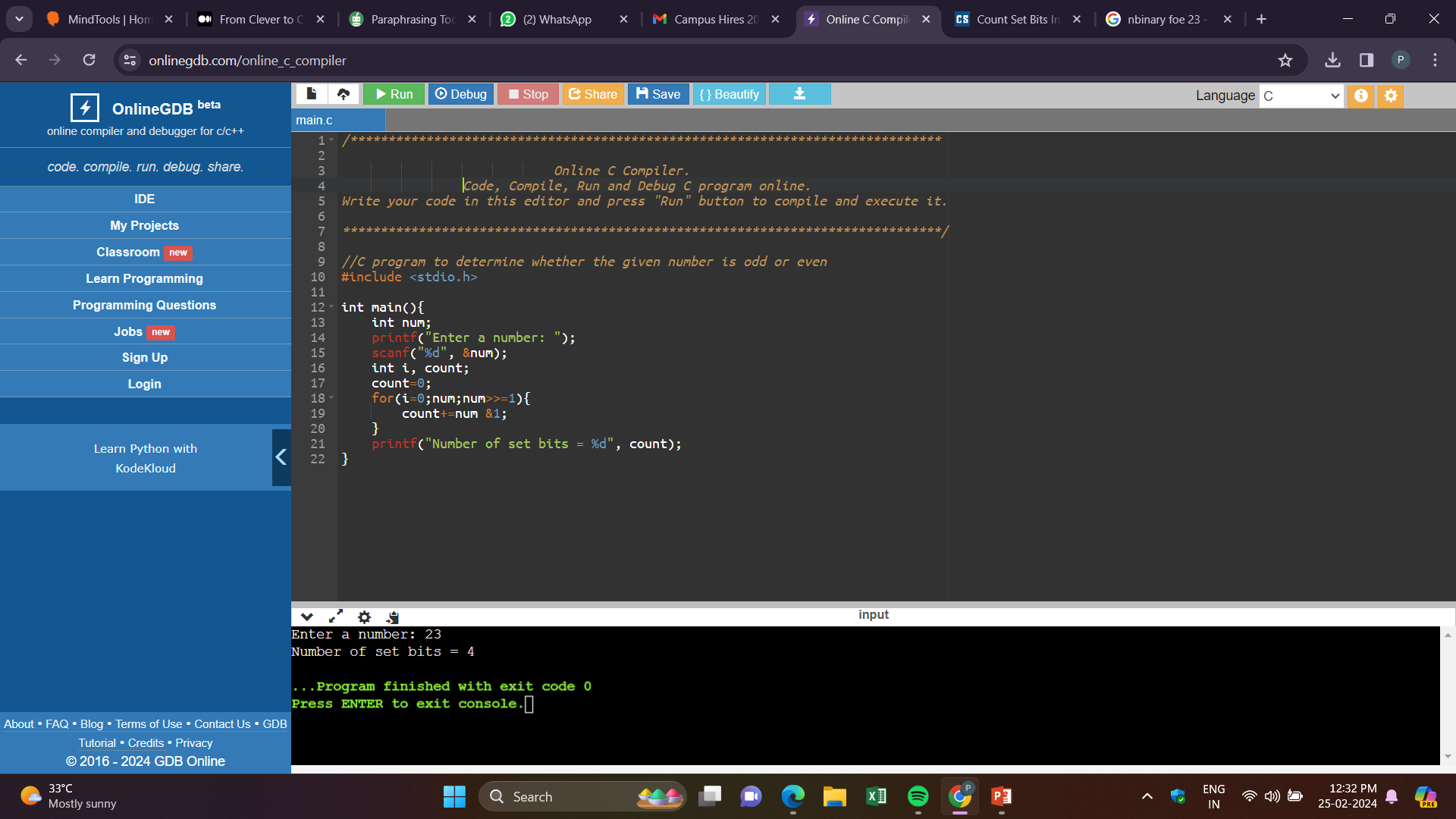
2. Write a C program to count the number of bits set in a number.

**Input:**

144

**Output:**

Count of Set bits: 2



3. Write a C program to swap two numbers. Use a function pointer to do this operation.

**Input:**

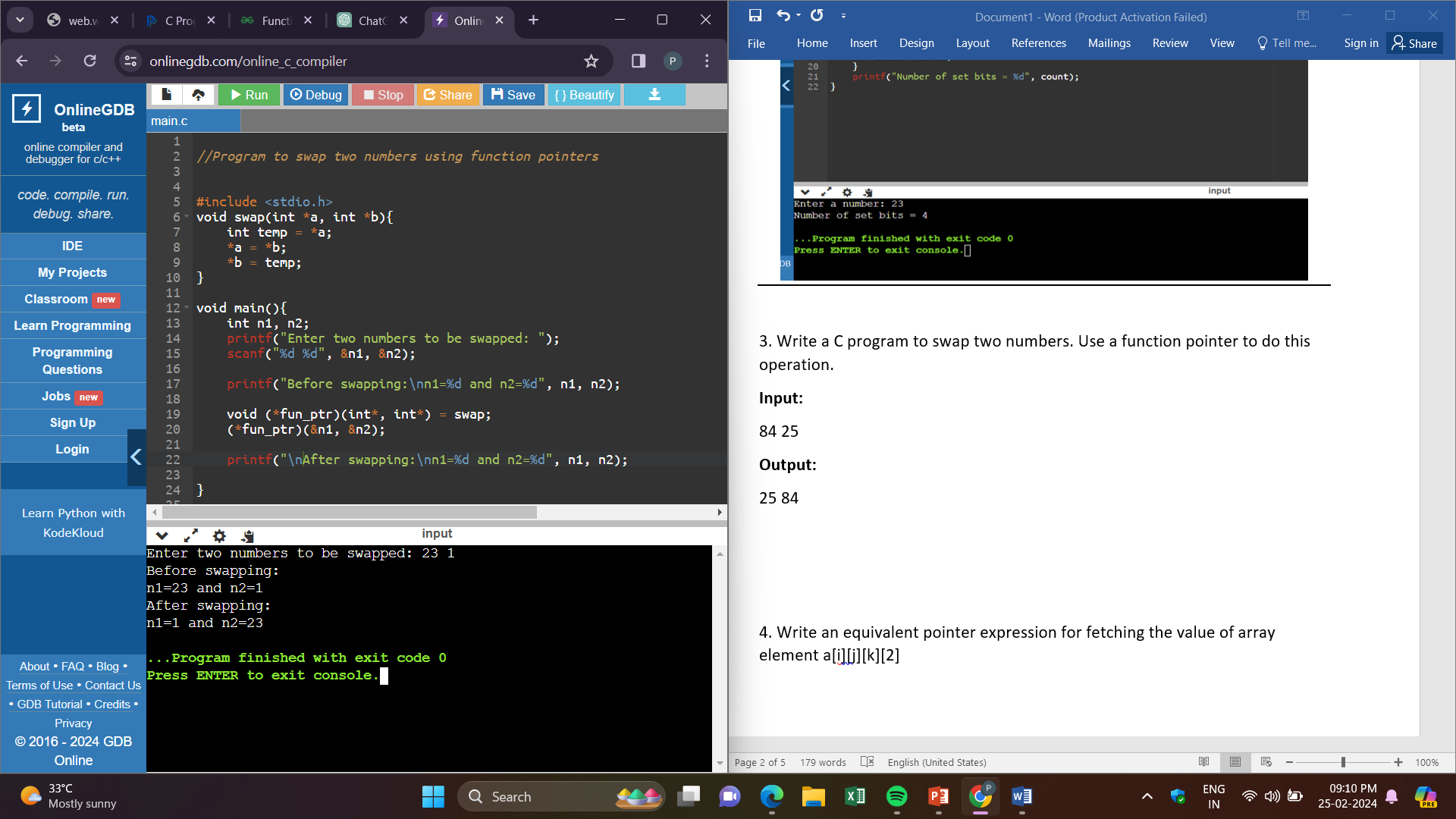
84 25

**Output:**

25 84

Solution:

Function pointers are used to point to a piece of code i.e., a function and not to data like normal pointers. It stores the start of the executable code.



4. Write an equivalent pointer expression for fetching the value of array element a[i][j][k][2]

Answer:

The equivalent pointer expression for a[i][j][k][2] is \*(\*(\*(\*(a+i)+j)+k)+2).

5. Write a C program to Multiply two matrix (n\*n) using pointers.

**Input:  Output:**

Size of Row: 3  Product:

Size of Column: 3  48 39 30

Matrix 1:  102 84 66

2 3 4  129 111 93

5 6 7

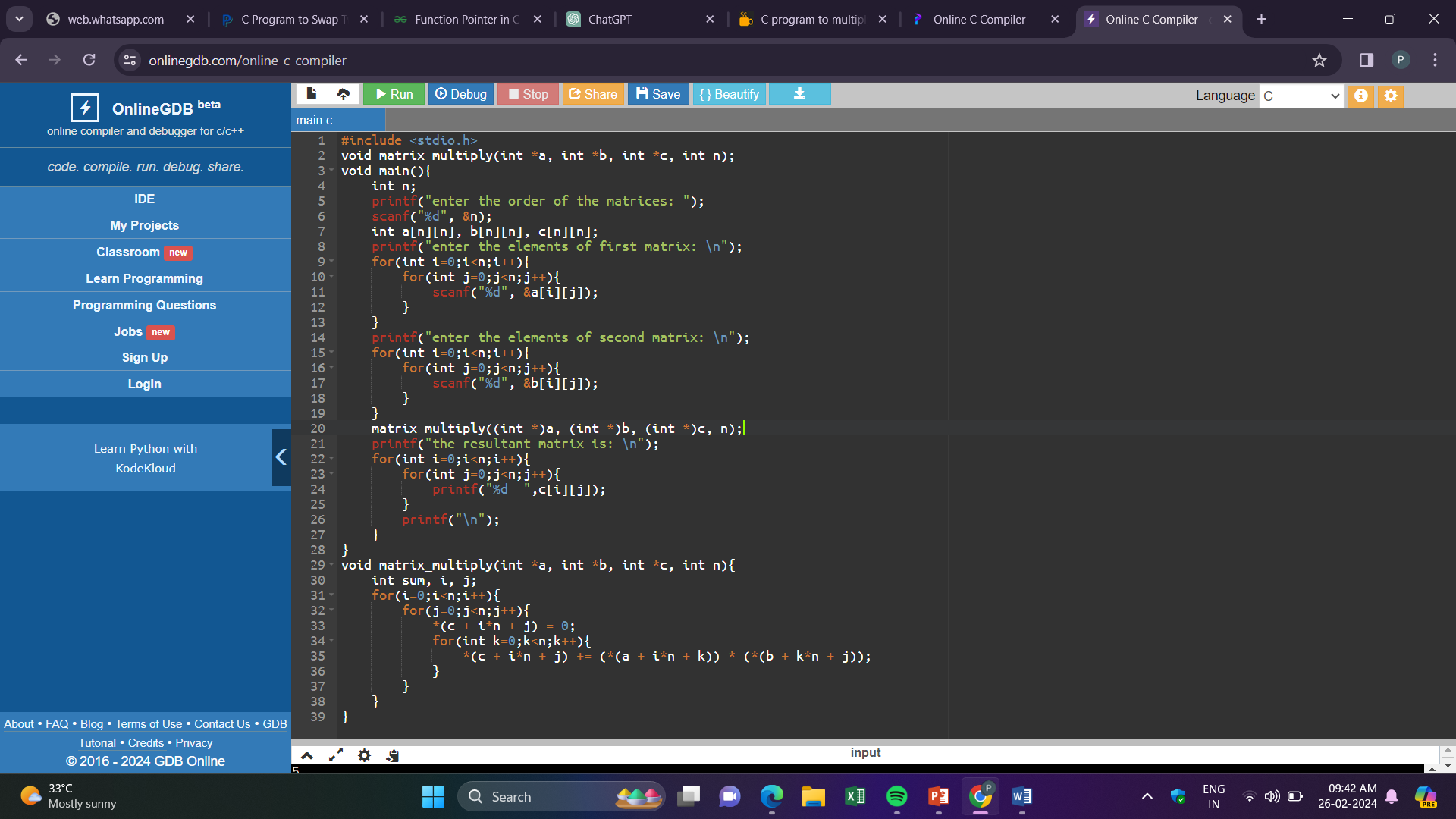
8 9 1

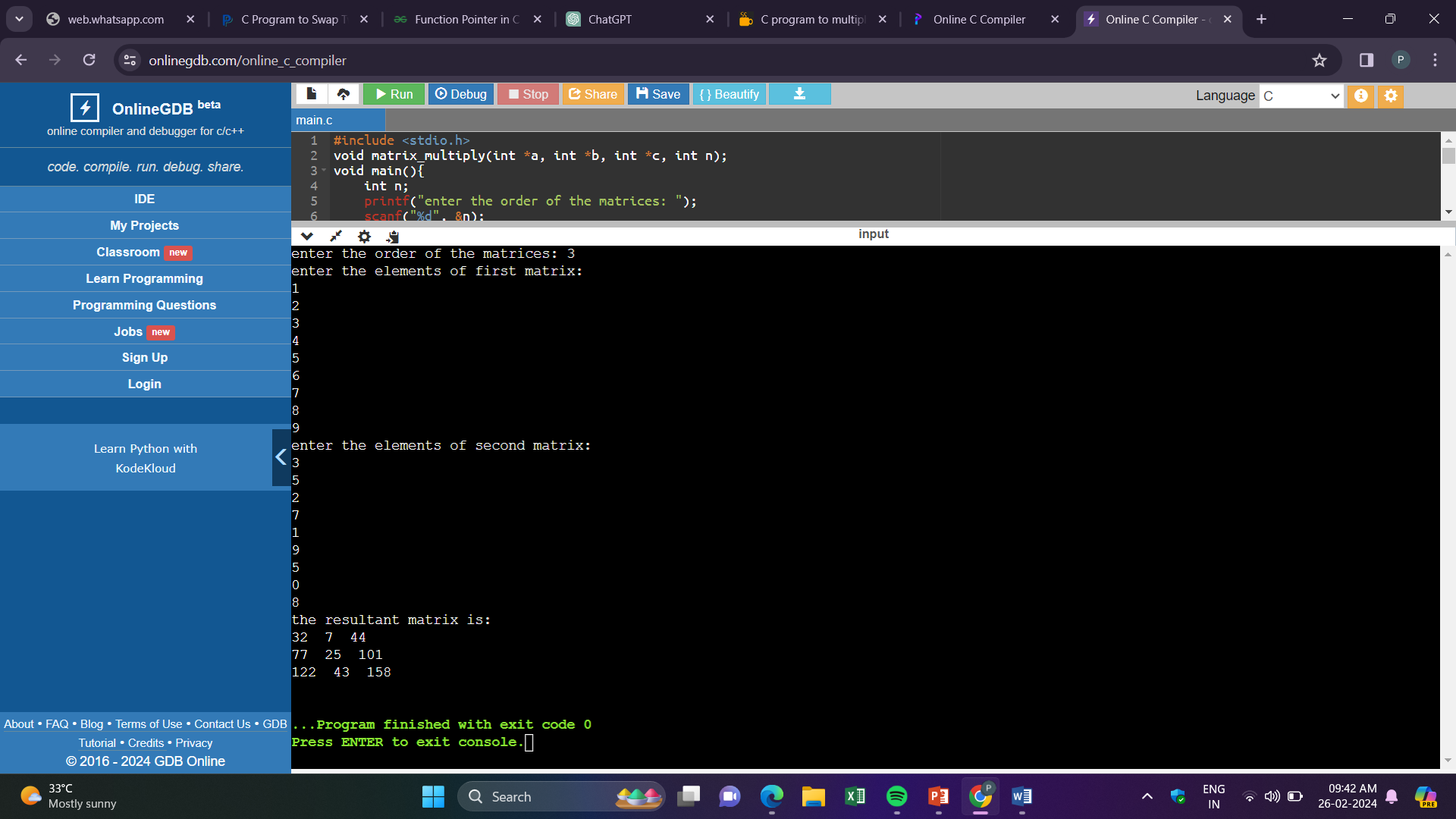
Matrix 2:

9 8 7

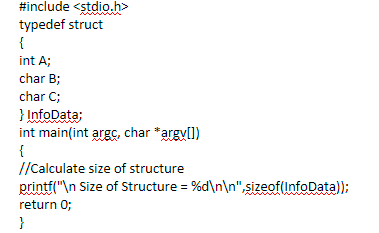
6 5 4

3 2 1





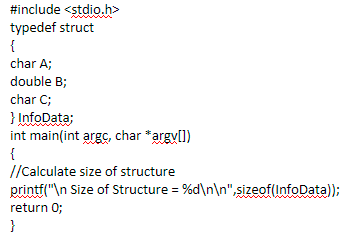
6. Find the output of the following // Consider the compiler is 32-bit machine



OUTPUT: Size of Structure = 8

EXPLANATION: The structure occupies contiguous memory locations. According to our idea, this structure should occupy 6 bytes of memory i.e., 4 bytes for integer, 1 byte for char and then 1 byte for char. But due to structure padding, and since the compiler is 32 bit, last 2 bytes of second word is empty. Thus int occupies one word (4 bytes) and char B and C occupies two bytes of second word and the remaining two bytes are empty.

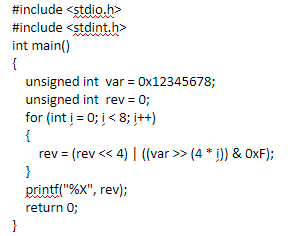
7. Find the output of the following // Consider the compiler is 32-bit machine



OUTPUT: Size of Structure = 16 (for 32 bit compiler)

EXPLANATION: The structure occupies contiguous memory locations. According to our idea, this structure should occupy 10 bytes of memory i.e., 1 byte for char A, 8 bytes for double C and then 1 byte for char C. But due to structure padding, and since the compiler is 32 bit (1 word = 4 bytes), char A occupies one byte in the first word. Since double should be allocated 8 bytes, it occupies second and third word which sums up to 12 bytes. Then char C occupies fourth word (1 byte for char and next 3 bytes are empty for padding). Thus the size of the structure is 16 bytes. If it is in 64 bit compiler, size would be 24 bytes.

8. Find the output of the following // Consider the compiler is 32-bit machine



OUTPUT: 87654321