**ASSIGNMENT-1**

**PROGRAM-1**

WRITE A C PROGRAM TO FIND THE SUM OF ELEMENTS IN AN ARRAY USING POINTERS

#include <stdio.h>

int main() {

int a[16], size, sum = 0, i;

int \*ptr;

printf("enter the size of an array: ");

scanf("%d", &size);

printf("enter the elements of an array: \n");

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

printf("enter a [%d] element=", i);

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

}

ptr = a;

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

sum = sum + \*ptr;

ptr++;

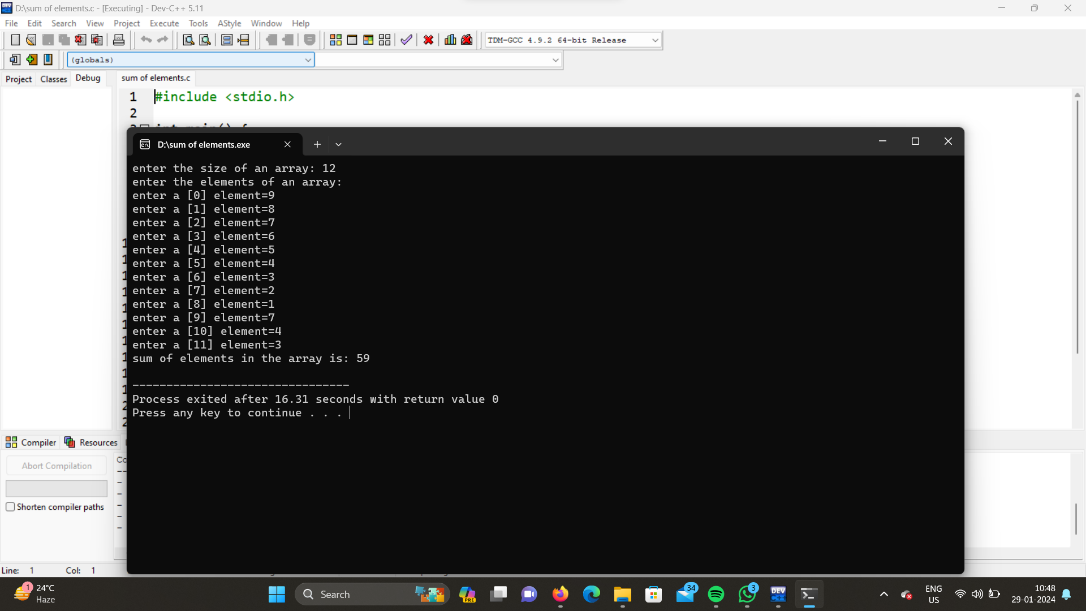
}

printf("sum of elements in the array is: %d\n", sum);

return 0;

}

OUTPUT:



**PROGRAM-2**

WRITE A C PROGRAM TO SWAP THE VALUES OF TWO INTEGERS USING POINTERS

#include <stdio.h>

int main() {

int a = 65;

int b = 54;

int c;

Printf("Before swapping, a is %d and b is %d\n", a, b);

c = a;

a = b;

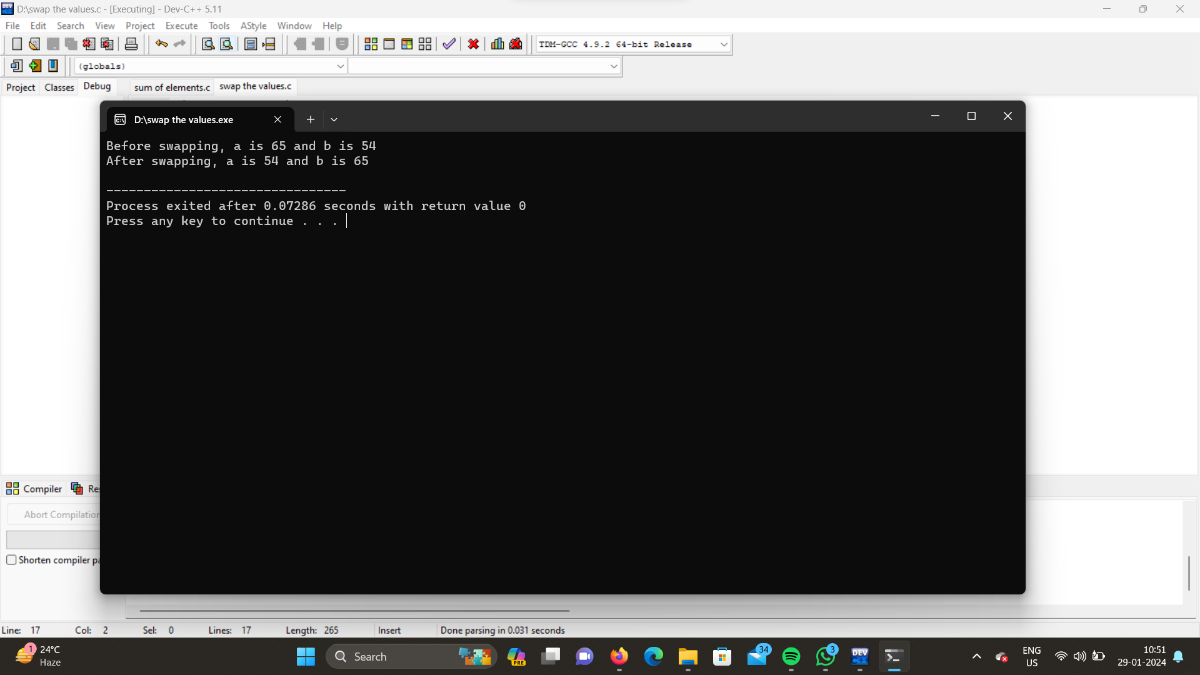
b = c;

printf("After swapping, a is %d and b is %d\n", a, b);

return 0;

}

OUTPUT:



**PROGRAM-3**

WRITE A C PROGRAM TO REVERSE A STRING USING POINTERS

#include <stdio.h>

#include <string.h>

void reverse(char \*str) {

char \*start = str;

char \*end = str + strlen(str) - 1;

char temp;

while (end > start) {

temp = \*start;

\*start = \*end;

\*end = temp;

start++;

end--;

}

}

int main() {

char str[] = "reverse string";

printf("Original string: %s\n", str);

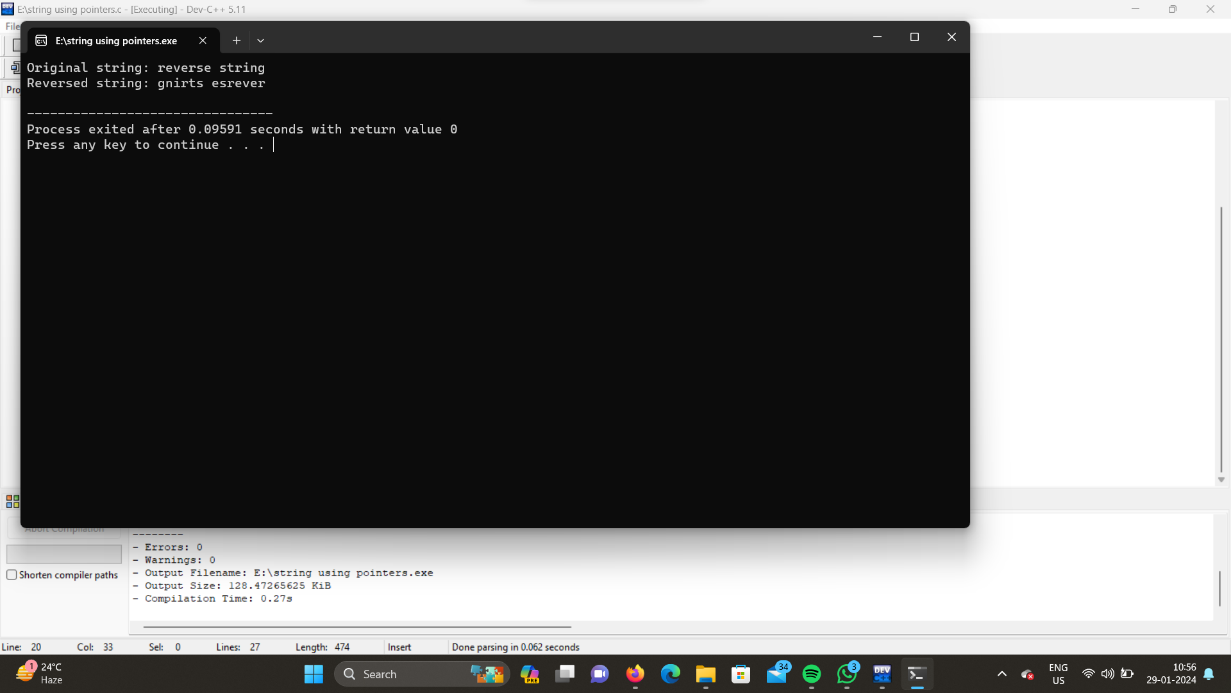
reverse(str);

printf("Reversed string: %s\n", str);

return 0;

}

OUTPUT:



**PROGRAM-4**

**WRITE A C PROGRAM TO CALCULATE THE POWER OF A NUMBER USING POINTERS TO FUNCTIONS**

#include <stdio.h>

int main() {

int base, exp;

long double result = 1.0;

printf("Enter a base number: ");

scanf("%d", &base);

printf("Enter an exponent: ");

scanf("%d", &exp);

while (exp != 0) {

result \*= base;

--exp;

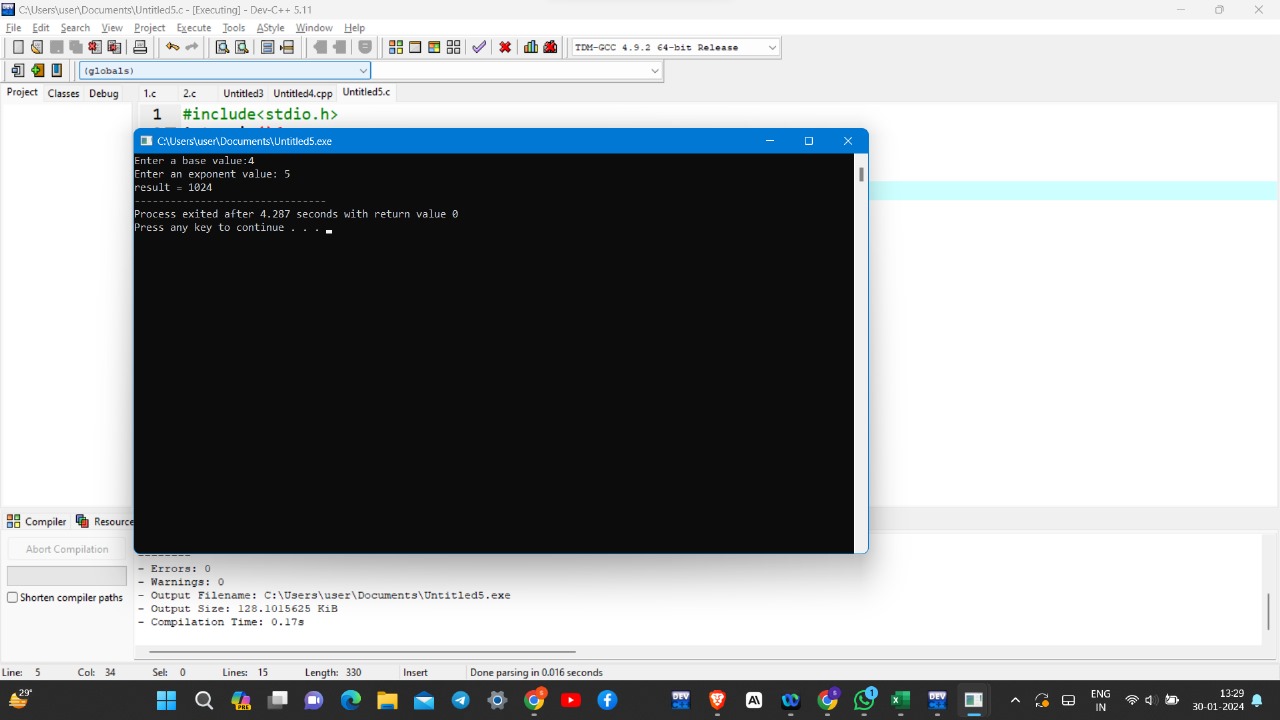
}

printf("Answer = %.0Lf", result);

return 0;

}

OUTPUT:



**PROGRAM-5**

**WRITE A C PROGRAM THAT DYNAMICALLY ALLOCATES MEMORY FOR A 2D ARRAY BASED ON USER INPUT**

#include <stdio.h>

#include <stdlib.h>

int main()

{

int rows, cols;

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

scanf("%d", &rows);

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

scanf("%d", &cols);

int \*\*arr = (int \*\*)malloc(rows \* sizeof(int \*));

if (arr == NULL) {

printf("Memory allocation failed.\n");

return 1;

}

for (int i = 0; i < rows; i++) {

arr[i] = (int \*)calloc(cols, sizeof(int));

if (arr[i] == NULL) {

printf("Memory allocation failed.\n");

return 1;

}

}

printf("Enter the elements of the 2D array:\n");

for (int i = 0; i < rows; i++) {

for (int j = 0; j < cols; j++) {

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

}

}

printf("\nThe entered 2D array is:\n");

for (int i = 0; i < rows; i++) {

for (int j = 0; j < cols; j++) {

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

}

printf("\n");

}

for (int i = 0; i < rows; i++) {

free(arr[i]);

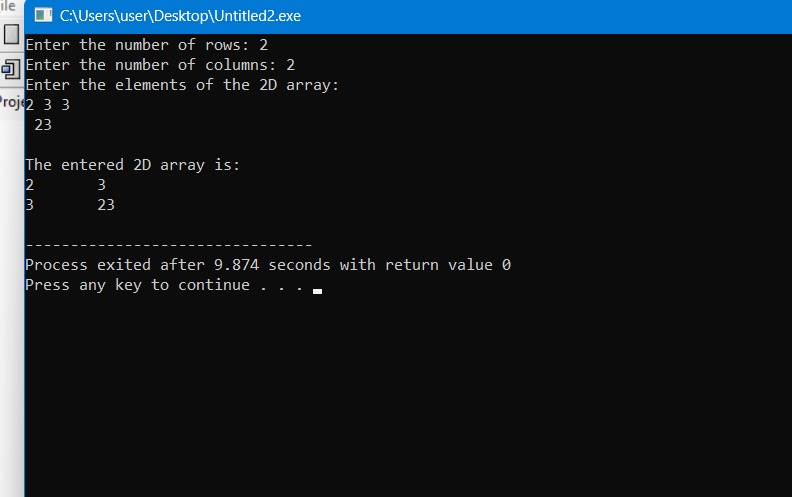
}

free(arr);

return 0;

}

OUTPUT:



**THE END**