

## 1.C program to display prime numbers between intervals.

The image shows the Dev-C++ IDE with a C program to find prime numbers between two intervals. The program is titled "prime number between intervals.cpp" and is being executed. The output window shows the program's execution, displaying the input values (12 and 30) and the prime numbers found (13, 17, 19, 23, 29).

**Code in prime number between intervals.cpp:**

```
1 #include <stdio.h>
2 int checkPrimeNumber(int n);
3 int main() {
4     int n1, n2, i, flag;
5     printf("Enter two positive integers: ");
6     scanf("%d %d", &n1, &n2);
7     if (n1 > n2) {
8         n1 = n1 + n2;
9         n2 = n1 - n2;
10        n1 = n1 - n2;
11    }
12    printf("Prime numbers between %d and %d are: ", n1, n2);
13    for (i = n1 + 1; i < n2; ++i) {
14        flag = checkPrimeNumber(i);
15
16        if (flag == 1) {
17            printf("%d ", i);
18        }
19    }
20    return 0;
21 }
22 int checkPrimeNumber(int n) {
23     int j, flag = 1;
24     for (j = 2; j <= n / 2; ++j) {
25         if (n % j == 0) {
```

**Execution Output:**

```
C:\Users\vish\OneDrive\Desktop\c++ programs\prime number between intervals.exe
Enter two positive integers: 12
30
Prime numbers between 12 and 30 are: 13 17 19 23 29
-----
Process exited after 10.82 seconds with return value 0
Press any key to continue . . .
```

**Compilation Results:**

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\vish\OneDrive\Desktop\c++ programs\prime number between intervals.exe
- Output Size: 129.1748046875 KiB
- Compilation Time: 0.55s
```

Line: 24 Col: 33 Sel: 0 Lines: 31 Length: 616 Insert Done parsing in 0.109 seconds

2. check whether the number can be expressed as sum of two prime numbers.

The screenshot shows the Dev-C++ IDE with the file `C:\Users\vish\OneDrive\Desktop\c++ programs\sum of two prime numbers.c` open. The code is as follows:

```
1 #include <stdio.h>
2 int checkPrime(int n);
3 int main() {
4     int n, i, flag = 0;
5     printf("Enter a positive integer: ");
6     scanf("%d", &n);
7     for (i = 2; i <= n / 2; ++i) {
8         if (checkPrime(i) == 1) {
9             if (checkPrime(n - i) == 1) {
10                 printf("%d = %d + %d\n", n, i, n - i);
11                 flag = 1;
12             }
13         }
14     }
15     if (flag == 0)
16         printf("%d cannot be expressed as the sum of two prime numbers.", n);
17     return 0;
18 }
19 int checkPrime(int n) {
20     int i, isPrime = 1;
21     if (n == 0 || n == 1) {
22         isPrime = 0;
23     }
24     else {
25         for(i = 2; i <= n/2; ++i) {
```

The output window shows the execution results for the input 34:

```
Enter a positive integer: 34
34 = 3 + 31
34 = 5 + 29
34 = 11 + 23
34 = 17 + 17

-----
Process exited after 2.418 seconds with return value 0
Press any key to continue . . .
```

The compilation results at the bottom show:

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\vish\OneDrive\Desktop\c++ programs\sum of two prime numbers.exe
- Output Size: 129.15625 KiB
- Compilation Time: 0.61s
```

### 3.LCM of two numbers.

The image shows a screenshot of the Dev-C++ 5.11 IDE. The main window displays the source code for a program to calculate the Least Common Multiple (LCM) of two numbers. The code is as follows:

```
1  #include <stdio.h>
2
3  int main() {
4
5      int n1, n2, max;
6
7      printf("Enter two positive integers: ");
8      scanf("%d %d", &n1, &n2);
9      max = (n1 > n2) ? n1 : n2;
10
11     while (1) {
12         if ((max % n1 == 0) && (max % n2 == 0)) {
13             printf("The LCM of %d and %d is %d.", n1, n2, max);
14             break;
15         }
16         ++max;
17     }
18     return 0;
19 }
```

Overlaid on the code editor is a black console window titled "C:\Users\vish\OneDrive\Desktop\c++ programs\LCM.exe". It shows the program's execution:

```
Enter two positive integers: 45
76
The LCM of 45 and 76 is 3420.
-----
Process exited after 6.252 seconds with return value 0
Press any key to continue . . .
```

At the bottom of the IDE, the "Compiler" tab is active, showing the "Compilation results...":

```
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\vish\OneDrive\Desktop\c++ programs\LCM.exe
- Output Size: 128.1015625 KiB
- Compilation Time: 0.17s
```

The status bar at the very bottom indicates "Line: 8 Col: 30 Sel: 0 Lines: 19 Length: 378 Insert Done parsing in 0.016 seconds". The Windows taskbar is visible at the bottom of the screen.

## 4.GCD of two numbers.

The image shows the Dev-C++ IDE interface. The main editor window displays the source code for a C++ program to calculate the Greatest Common Divisor (GCD) of two numbers. The code is as follows:

```
1  #include <stdio.h>
2  int hcf(int n1, int n2);
3  int main() {
4      int n1, n2;
5      printf("Enter two positive integers: ");
6      scanf("%d %d", &n1, &n2);
7      printf("G.C.D of %d and %d is %d.", n1, n2, hcf(n1, n2));
8      return 0;
9  }
10
11 int hcf(int n1, int n2) {
12     if (n2 != 0)
13         return hcf(n2, n1 % n2);
14     else
15         return n1;
16 }
17
```

Overlaid on the IDE is a terminal window titled "C:\Users\vvish\OneDrive\Desktop\c++ programs\GCD.exe". It shows the program's execution with the following output:

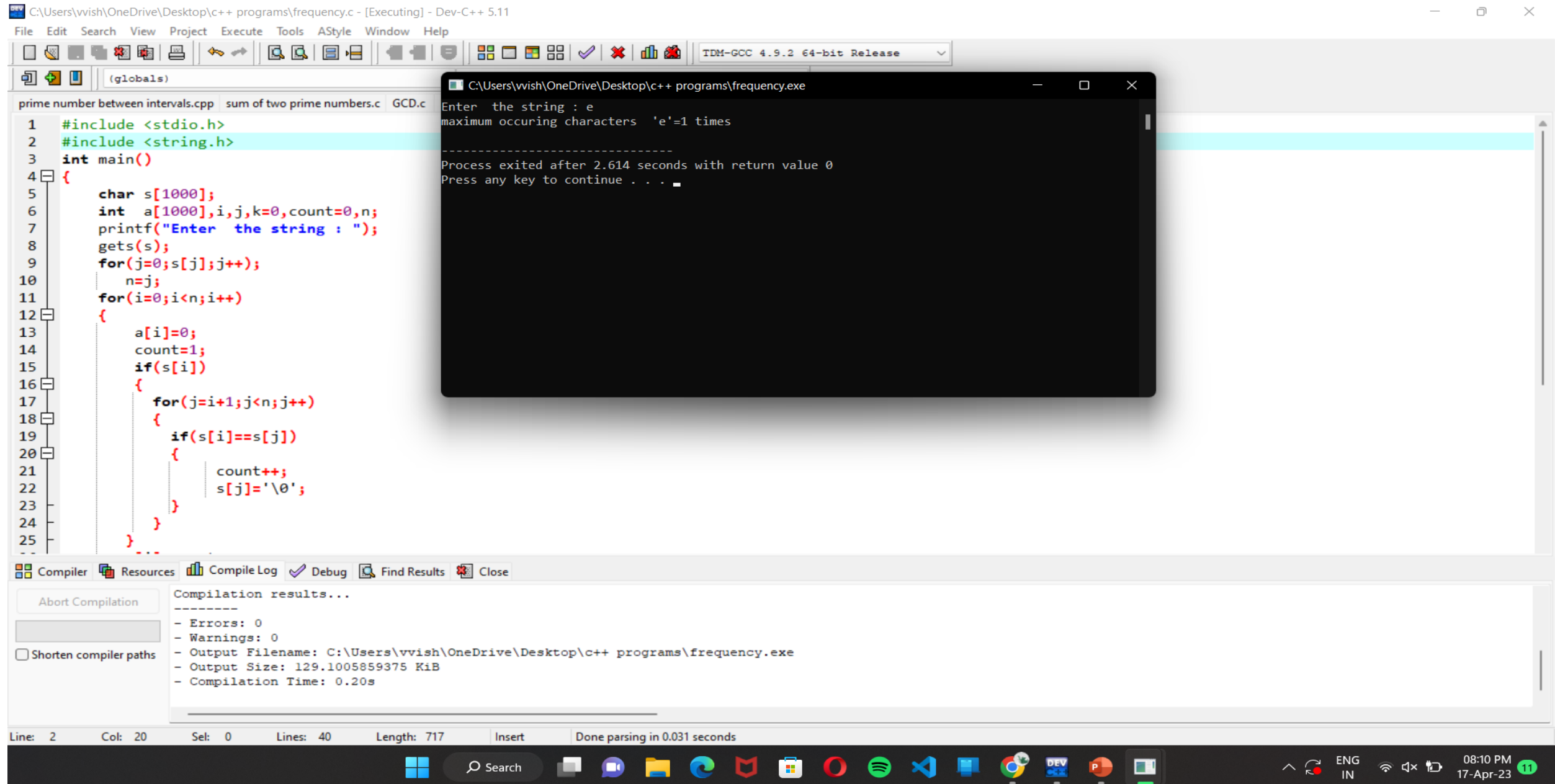
```
Enter two positive integers: 34
65
G.C.D of 34 and 65 is 1.
-----
Process exited after 13.59 seconds with return value 0
Press any key to continue . . .
```

At the bottom of the IDE, the "Compiler" window shows the compilation results:

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\vvish\OneDrive\Desktop\c++ programs\GCD.exe
- Output Size: 128.619140625 KiB
- Compilation Time: 0.17s
```

The status bar at the bottom indicates the current cursor position: Line: 17, Col: 1, Sel: 0, Lines: 17, Length: 351. It also shows "Insert" mode and "Done parsing in 0 seconds". The Windows taskbar is visible at the very bottom, showing the time as 08:02 PM on 17-Apr-23.

## 5.C program to find frequency character in a string.



The image shows a C++ IDE (Dev-C++ 5.11) with a project named "frequency.c". The code is as follows:

```
1 #include <stdio.h>
2 #include <string.h>
3 int main()
4 {
5     char s[1000];
6     int a[1000], i, j, k=0, count=0, n;
7     printf("Enter the string : ");
8     gets(s);
9     for(j=0; s[j]; j++);
10    n=j;
11    for(i=0; i<n; i++)
12    {
13        a[i]=0;
14        count=1;
15        if(s[i])
16        {
17            for(j=i+1; j<n; j++)
18            {
19                if(s[i]==s[j])
20                {
21                    count++;
22                    s[j]='\0';
23                }
24            }
25        }
26    }
27 }
```

The program is being executed, and the output window shows the following text:

```
Enter the string : e
maximum occurring characters 'e'=1 times

-----
Process exited after 2.614 seconds with return value 0
Press any key to continue . . .
```

The compilation results are also shown in the bottom panel:

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\vish\OneDrive\Desktop\c++ programs\frequency.exe
- Output Size: 129.1005859375 KiB
- Compilation Time: 0.20s
```

The status bar at the bottom indicates the current line is 2, column is 20, and the selection is 0. The total lines in the file are 40, and the length is 717. The IDE also shows the Windows taskbar at the bottom with the time 08:10 PM on 17-Apr-23.

## 6.c program for Anagram program.

The screenshot displays the Dev-C++ IDE with a C++ program for checking anagrams. The program is named `anagram.c` and is located at `C:\Users\vish\OneDrive\Desktop\c++ programs\anagram.c`. The IDE is running version 5.11 of Dev-C++ with the TDM-GCC 4.9.2 64-bit Release compiler.

The source code for `anagram.c` is as follows:

```
1 #include <stdio.h>
2
3 int find_anagram(char [], char []);
4
5 int main()
6 {
7     char array1[100], array2[100];
8     int flag;
9
10    printf("Enter the string\n");
11    gets(array1);
12    printf("Enter another string\n");
13    gets(array2);
14    flag = find_anagram(array1, array2);
15    if (flag == 1)
16        printf("%s and %s are anagrams.\n", array1, array2);
17    else
18        printf(" %s and %s are not anagrams.\n", array1, array2);
19    return 0;
20 }
21
22 int find_anagram(char array1[], char array2[])
23 {
24     int num1[26] = {0}, num2[26] = {0}, i = 0;
25
26     while (array1[i] != '\0')
```

The program is compiled and executed. The output window shows the following text:

```
C:\Users\vish\OneDrive\Desktop\c++ programs\anagram.exe
Enter the string
hectare
Enter another string
teacher
hectare and teacher are anagrams.

-----
Process exited after 17.76 seconds with return value 0
Press any key to continue . . .
```

The compilation results are shown in the bottom panel:

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\vish\OneDrive\Desktop\c++ programs\anagram.exe
- Output Size: 129.7998046875 KiB
- Compilation Time: 0.17s
```

The status bar at the bottom indicates the current line is 20, column is 2, and the program is done parsing in 0.016 seconds.

## 7.C program to find sum of ascii value of all ascii character.

The image shows the Dev-C++ IDE interface. The main window displays a C program that calculates the sum of ASCII values for a given string. The program is titled "sum of ascii values.c" and is located at "C:\Users\vish\OneDrive\Desktop\c++ programs\sum of ascii values.c". The code is as follows:

```
1 #include <stdio.h>
2 #include <string.h>
3
4 void main()
5 {
6     int sum = 0, i, len;
7     char string1[100];
8
9     printf("Enter the string : ");
10    scanf("%[^\n]s", string1);
11    len = strlen(string1);
12    for (i = 0; i < len; i++)
13    {
14        sum = sum + string1[i];
15    }
16    printf("\nSum of all characters : %d ", sum);
17 }
```

The program is compiled and executed. The output window shows the following text:

```
C:\Users\vish\OneDrive\Desktop\c++ programs\sum of ascii values.exe
Enter the string : welcome to my world

Sum of all characters : 1853
-----
Process exited after 20.52 seconds with return value 30
Press any key to continue . . .
```

The compilation results window shows the following information:

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\vish\OneDrive\Desktop\c++ programs\sum of ascii values.exe
- Output Size: 128.123046875 KiB
- Compilation Time: 0.19s
```

The status bar at the bottom indicates the current line and column: Line: 17, Col: 2. The taskbar at the bottom shows the Windows logo, search bar, and various application icons.



## 8.C program to print all permutations.

C:\Users\vish\OneDrive\Desktop\c++ programs\permutations.c - [Executing] - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

(globals)  
prime number between intervals.cpp sum of two prime numbers.c GCD.c

```
1  #include <stdio.h>
2  #include <string.h>
3  void swap(char* x, char* y)
4  {
5      char temp;
6      temp = *x;
7      *x = *y;
8      *y = temp;
9  }
10 void permute(char* a, int l, int r)
11 {
12     int i;
13     if (l == r)
14         printf("%s\n", a);
15     else {
16         for (i = l; i <= r; i++) {
17             swap((a + l), (a + i));
18             permute(a, l + 1, r);
19             swap((a + l), (a + i));
20         }
21     }
22 }
23 int main()
24 {
25     char str[] = "ABC";
```

C:\Users\vish\OneDrive\Desktop\c++ programs\permutations.exe

ABC  
ACB  
BAC  
BCA  
CBA  
CAB

-----  
Process exited after 0.02761 seconds with return value 0  
Press any key to continue . . .

Compiler Resources Compile Log Debug Find Results Close

Abort Compilation

Compilation results...

-----  
- Errors: 0  
- Warnings: 0  
- Output Filename: C:\Users\vish\OneDrive\Desktop\c++ programs\permutations.exe  
- Output Size: 128.4296875 KiB  
- Compilation Time: 0.17s

Line: 22 Col: 2 Sel: 0 Lines: 30 Length: 447 Insert Done parsing in 0.016 seconds



Search



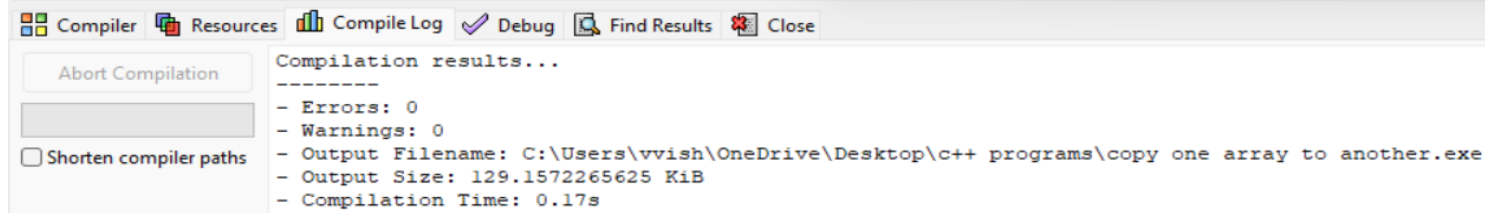
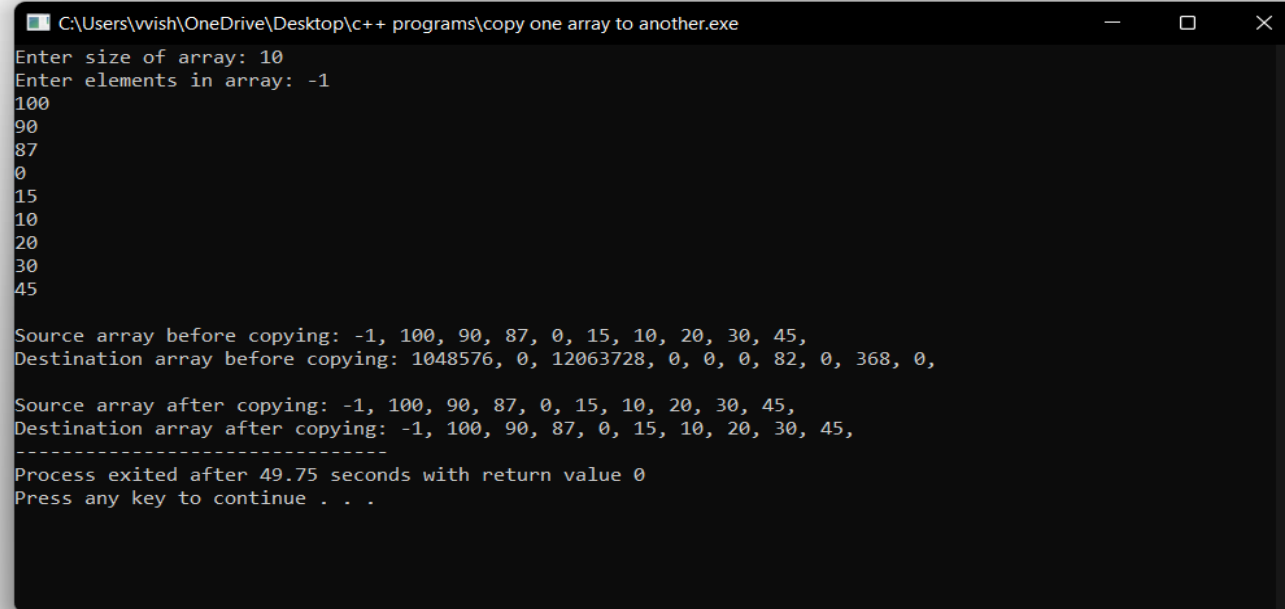
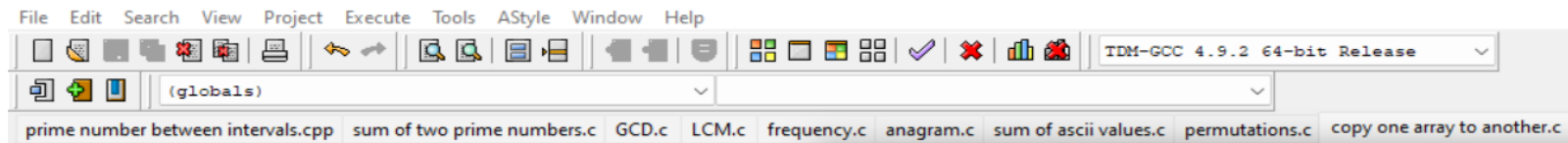
ENG  
IN

08:19 PM  
17-Apr-23

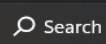


## 9.C program to copy one array element to another.

C:\Users\vish\OneDrive\Desktop\c++ programs\copy one array to another.c - [Executing] - Dev-C++ 5.11



Line: 43 Col: 11 Sel: 0 Lines: 48 Length: 1121 Insert Done parsing in 0.016 seconds



ENG  
IN

08:24 PM  
17-Apr-23

## 10.C program to reverse a string.

The image shows a C++ IDE with a source code editor, a compiler window, and a console window.

**Source Code (reverse string.c):**

```
1 #include <stdio.h>
2 #include <string.h>
3 void reverseString(char* str)
4 {
5     int l, i;
6     char *begin_ptr, *end_ptr, ch;
7     l = strlen(str);
8     begin_ptr = str;
9     end_ptr = str + l - 1;
10    for (i = 0; i < (l - 1) / 2; i++) {
11        ch = *end_ptr;
12        *end_ptr = *begin_ptr;
13        *begin_ptr = ch;
14        begin_ptr++;
15        end_ptr--;
16    }
17 }
18 int main()
19 {
20     char str[100] = "GeeksForGeeks";
21     printf("Enter a string: %s\n", str);
22     reverseString(str);
23     printf("Reverse of the string: %s\n", str);
24     return 0;
25 }
```

**Compiler Window:**

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\vish\OneDrive\Desktop\c++ programs\reverse string.exe
- Output Size: 128.462890625 KiB
- Compilation Time: 0.17s

**Console Window (reverse string.exe):**

```
Enter a string: GeeksForGeeks
Reverse of the string: skeeGroFskeeG
-----
Process exited after 0.02741 seconds with return value 0
Press any key to continue . . .
```

**IDE Status Bar:** Line: 23 Col: 48 Sel: 0 Lines: 26 Length: 496 Insert Done parsing in 0.015 seconds