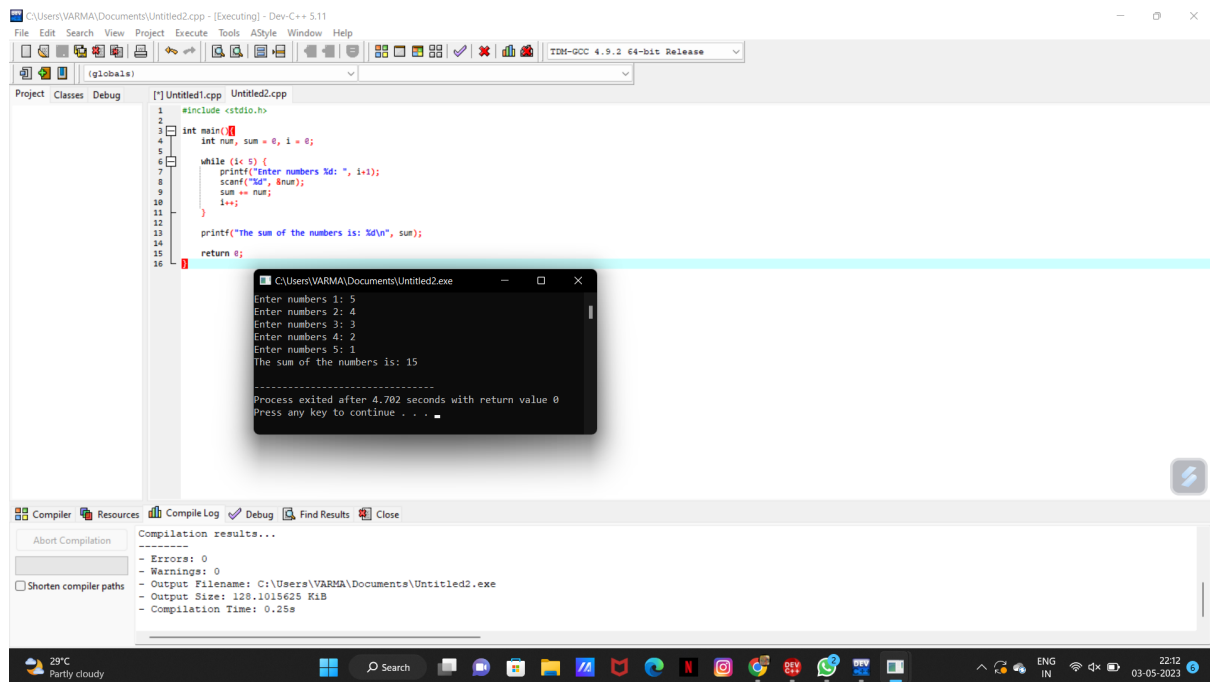


## 11.GET 5 NUMBERS FROM USER AND SUM IT USING WHILE LOOP



```
1 #include <stdio.h>
2
3 int main()
4 {
5     int num, sum = 0, i = 0;
6     while (i < 5) {
7         printf("Enter numbers %d: ", i+1);
8         scanf("%d", &num);
9         sum += num;
10        i++;
11    }
12    printf("The sum of the numbers is: %d\n", sum);
13
14    return 0;
15 }
```

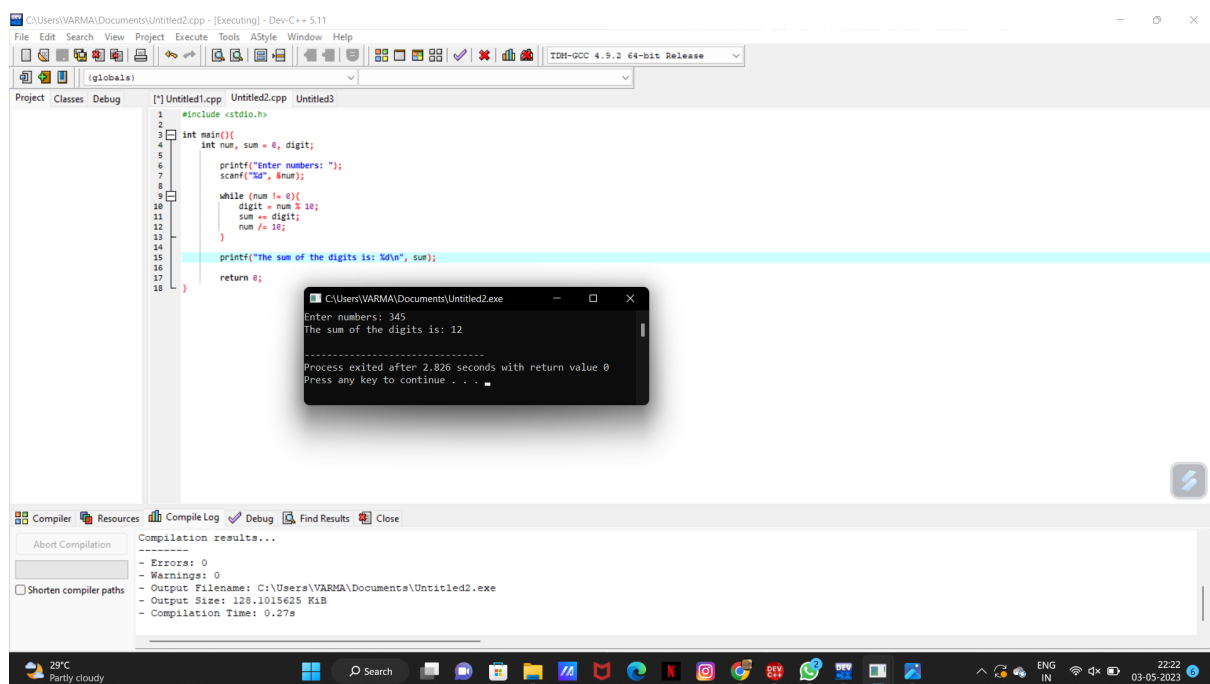
Enter numbers 1: 5  
Enter numbers 2: 4  
Enter numbers 3: 3  
Enter numbers 4: 2  
Enter numbers 5: 1  
The sum of the numbers is: 15

Process exited after 4.702 seconds with return value 0  
Press any key to continue . . .

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\VARMA\Documents\Untitled2.exe
- Output Size: 128.1015625 Kib
- Compilation Time: 0.25s

## 12. SUM OF DIGITS USING WHILE LOOP



```
1 #include <stdio.h>
2
3 int main()
4 {
5     int num, sum = 0, digit;
6     printf("Enter numbers: ");
7     scanf("%d", &num);
8     while (num != 0) {
9         digit = num % 10;
10        sum += digit;
11        num /= 10;
12    }
13    printf("The sum of the digits is: %d\n", sum);
14
15    return 0;
16 }
```

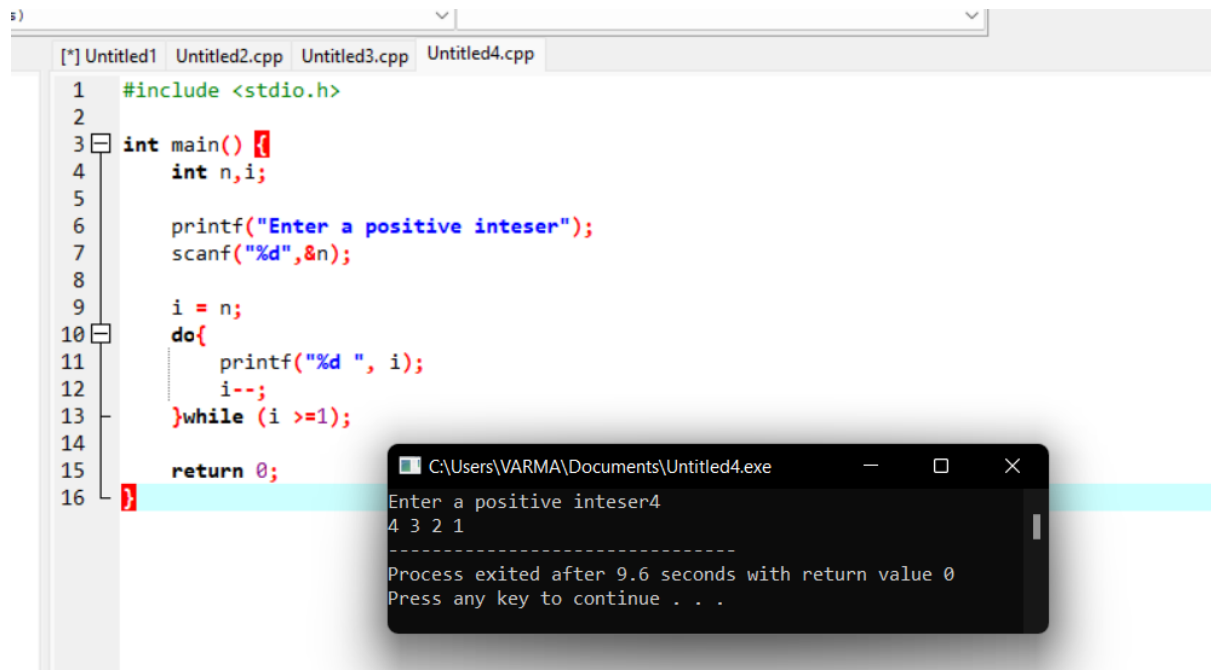
Enter numbers: 345  
The sum of the digits is: 12

Process exited after 2.826 seconds with return value 0  
Press any key to continue . . .

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\VARMA\Documents\Untitled2.exe
- Output Size: 128.1015625 Kib
- Compilation Time: 0.27s

### 13.PRINT THE FIRST N NUMBERS IN REVERSE USING DO VALUE

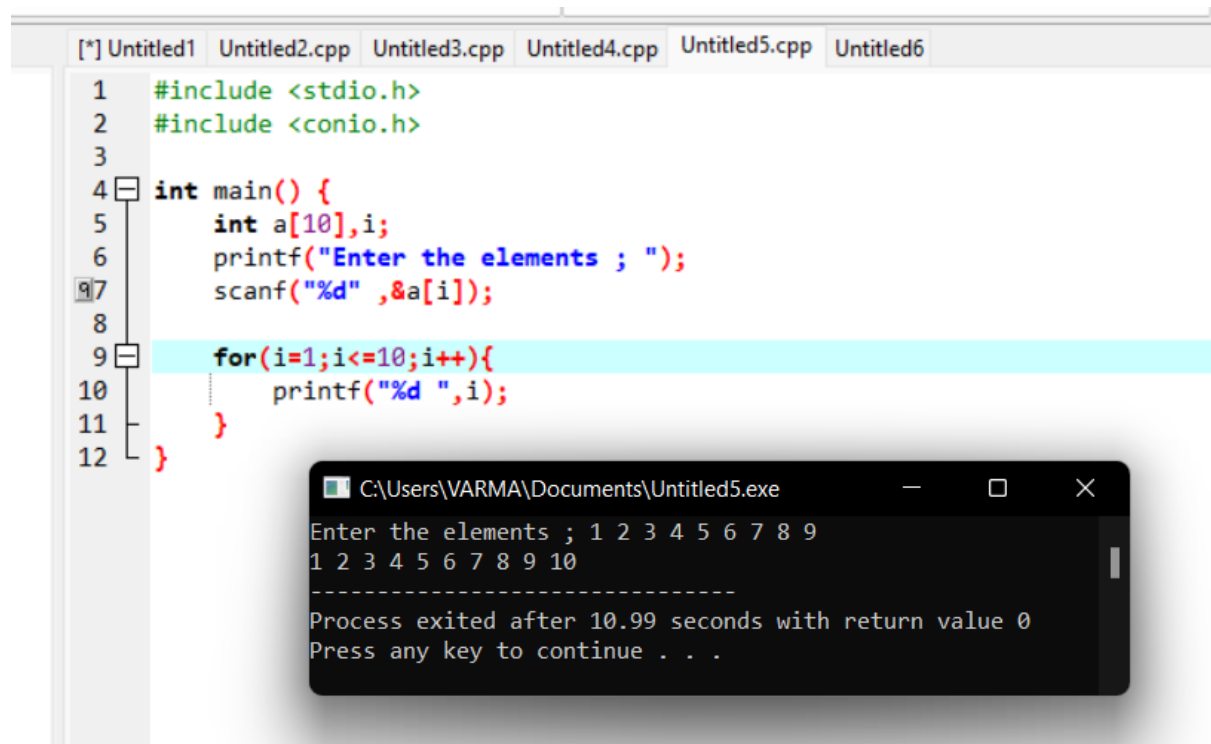


```
1  #include <stdio.h>
2
3  int main() {
4      int n,i;
5
6      printf("Enter a positive inteser");
7      scanf("%d",&n);
8
9      i = n;
10     do{
11         printf("%d ", i);
12         i--;
13     }while (i >=1);
14
15     return 0;
16 }
```

C:\Users\VARMA\Documents\Untitled4.exe

Enter a positive inteser4  
4 3 2 1  
-----  
Process exited after 9.6 seconds with return value 0  
Press any key to continue . . .

### 14. INNITIALIZATION OF ARRAY

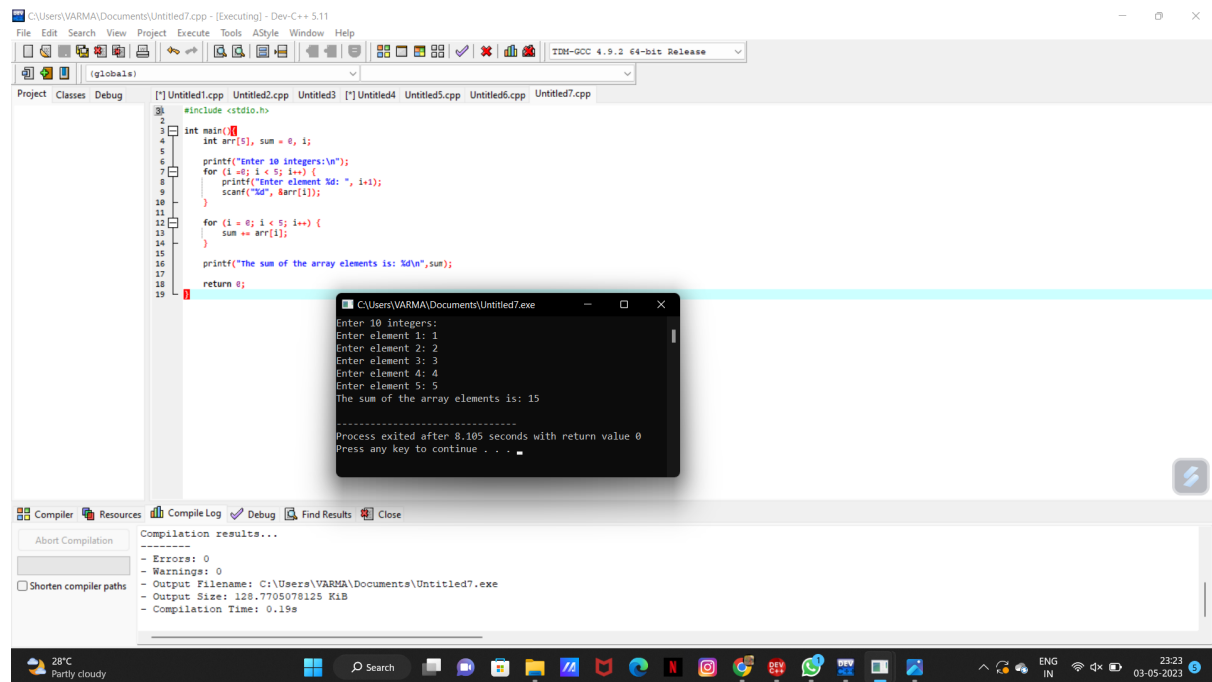


```
1  #include <stdio.h>
2  #include <conio.h>
3
4  int main() {
5      int a[10],i;
6      printf("Enter the elements ; ");
7      scanf("%d" ,&a[i]);
8
9      for(i=1;i<=10;i++){
10         printf("%d ",i);
11     }
12 }
```

C:\Users\VARMA\Documents\Untitled5.exe

Enter the elements ; 1 2 3 4 5 6 7 8 9  
1 2 3 4 5 6 7 8 9 10  
-----  
Process exited after 10.99 seconds with return value 0  
Press any key to continue . . .

## 15. SUM OF ARRAY ELEMENTS



The screenshot shows the Dev-C++ IDE with a C++ program to calculate the sum of 10 integers. The code is as follows:

```
#include <stdio.h>

int main()
{
    int arr[5], sum = 0, i;

    printf("Enter 10 integers:\n");
    for (i = 0; i < 5; i++) {
        printf("Enter element %d: ", i+1);
        scanf("%d", &arr[i]);
    }

    for (i = 0; i < 5; i++) {
        sum += arr[i];
    }

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

    return 0;
}
```

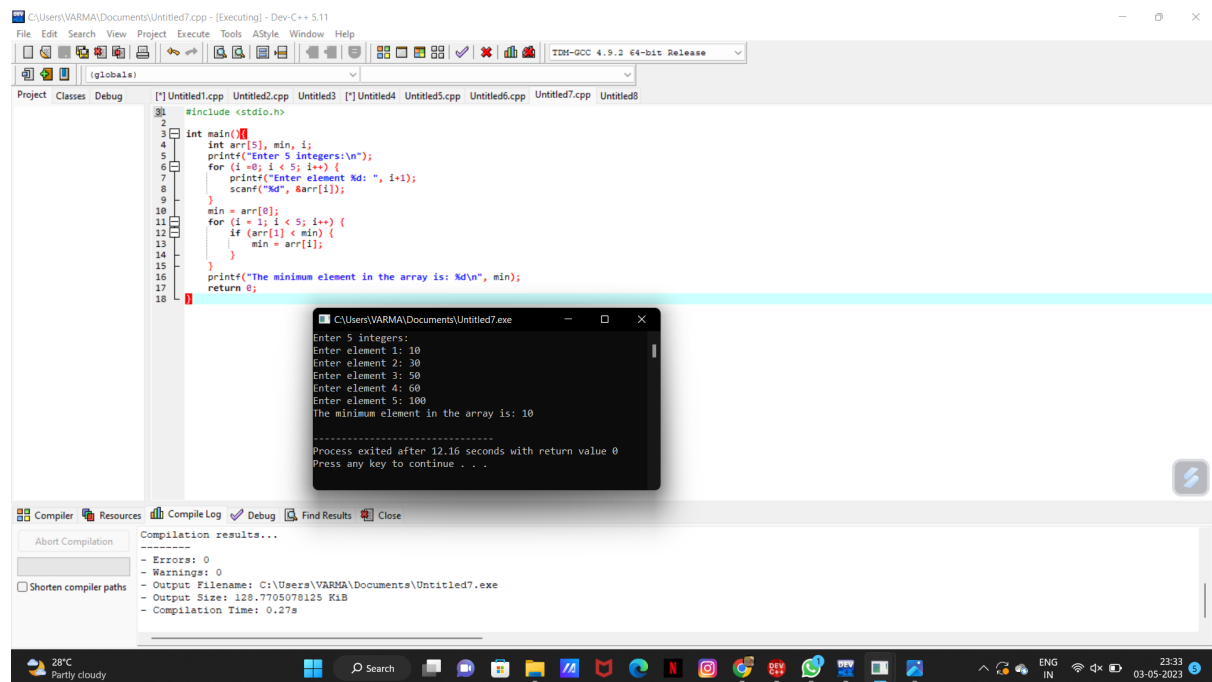
The execution output window shows the following text:

```
Enter 10 integers:
Enter element 1: 1
Enter element 2: 2
Enter element 3: 3
Enter element 4: 4
Enter element 5: 5
The sum of the array elements is: 15
.....
Process exited after 8.105 seconds with return value 0
Press any key to continue . . .
```

The compiler output window shows the following details:

```
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\VARMA\Documents\Untitled7.exe
- Output Size: 128.7705078125 KiB
- Compilation Time: 0.19s
```

## 16. MINIMUM ELEMENT IN ARRAY



The screenshot shows the Dev-C++ IDE with a C++ program to find the minimum element in an array of 5 integers. The code is as follows:

```
#include <stdio.h>

int main()
{
    int arr[5], min, i;

    printf("Enter 5 integers:\n");
    for (i = 0; i < 5; i++) {
        printf("Enter element %d: ", i+1);
        scanf("%d", &arr[i]);
    }

    min = arr[0];
    for (i = 1; i < 5; i++) {
        if (arr[i] < min) {
            min = arr[i];
        }
    }

    printf("The minimum element in the array is: %d\n", min);

    return 0;
}
```

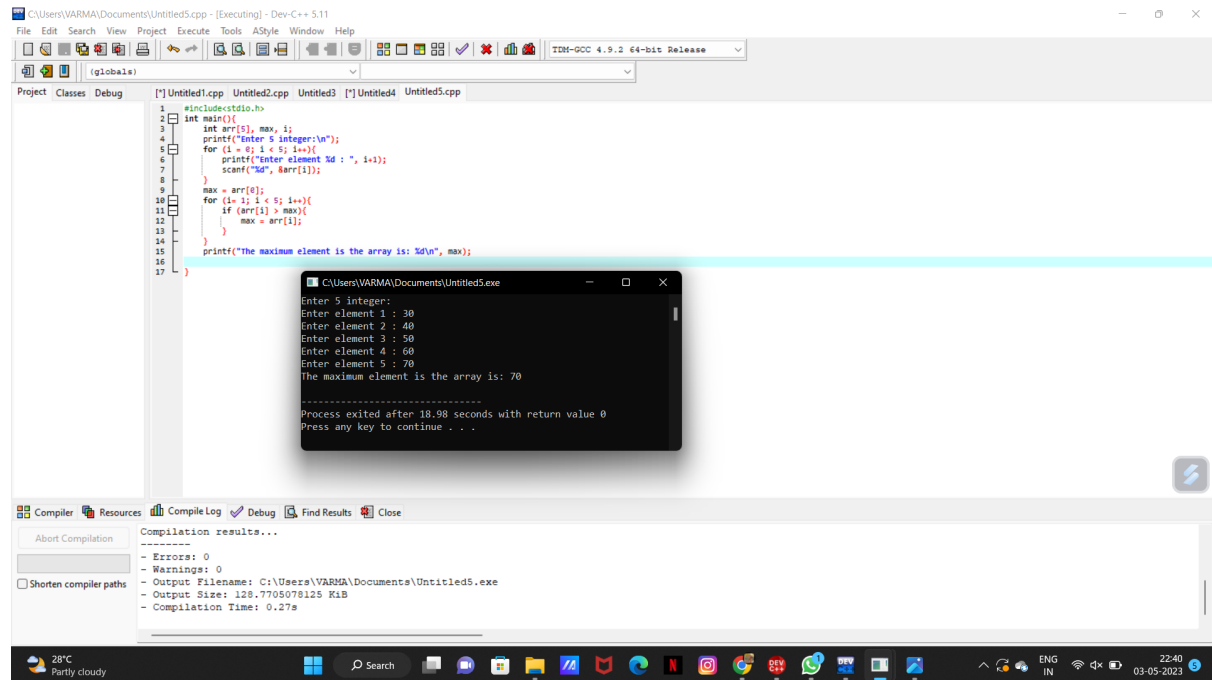
The execution output window shows the following text:

```
Enter 5 integers:
Enter element 1: 10
Enter element 2: 30
Enter element 3: 50
Enter element 4: 60
Enter element 5: 100
The minimum element in the array is: 10
.....
Process exited after 12.16 seconds with return value 0
Press any key to continue . . .
```

The compiler output window shows the following details:

```
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\VARMA\Documents\Untitled7.exe
- Output Size: 128.7705078125 KiB
- Compilation Time: 0.27s
```

## 17. MAXIMUM ELEMENT IN AN ARRAY

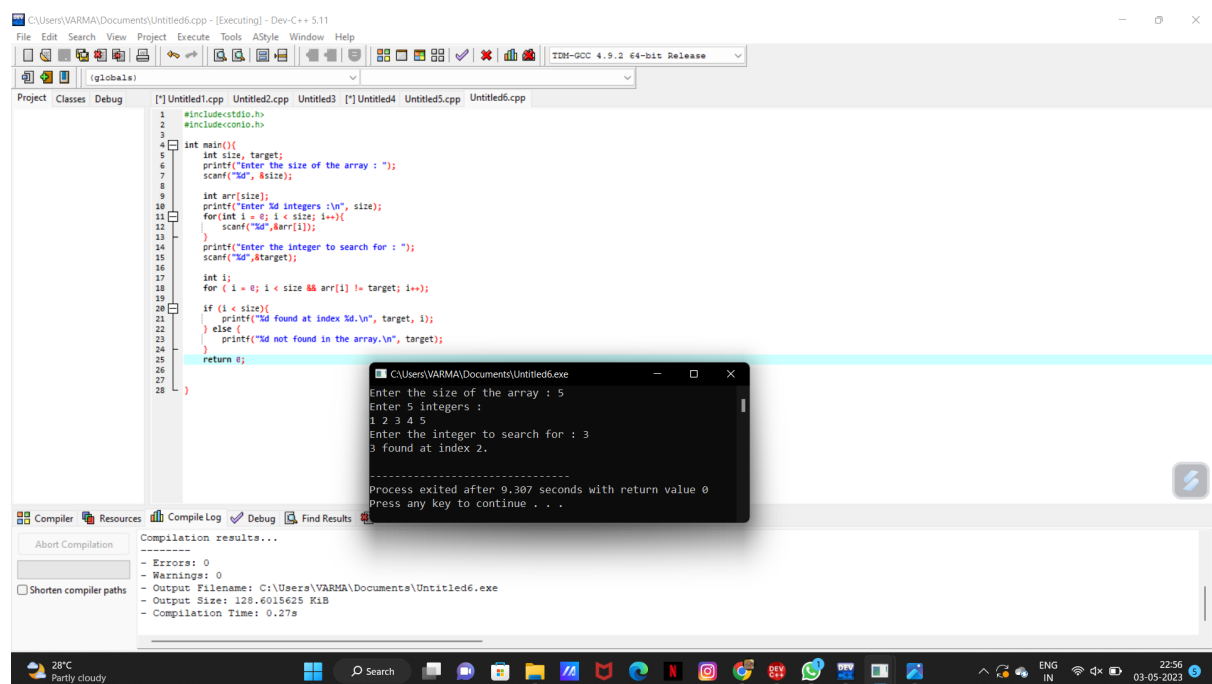


```
#include<stdio.h>
int main()
{
    int arr[5], max, i;
    printf("Enter 5 integer:\n");
    for (i = 0; i < 5; i++){
        printf("Enter element %d : ", i+1);
        scanf("%d", &arr[i]);
    }
    max = arr[0];
    for (i = 1; i < 5; i++){
        if (arr[i] > max){
            max = arr[i];
        }
    }
    printf("The maximum element is the array is: %d\n", max);
}
```

Enter 5 integer:  
Enter element 1 : 30  
Enter element 2 : 40  
Enter element 3 : 50  
Enter element 4 : 60  
Enter element 5 : 70  
The maximum element is the array is: 70  
-----  
Process exited after 18.98 seconds with return value 0  
Press any key to continue . . .

Compilation results...  
- Errors: 0  
- Warnings: 0  
- Output Filename: C:\Users\VARMA\Documents\Untitled5.exe  
- Output Size: 128.7705078125 KiB  
- Compilation Time: 0.27s

## 18. SEARCH ELEMENT IN ARRAY USING LINEAR SEARCH



```
#include<stdio.h>
#include<conio.h>
int main()
{
    int size, target;
    printf("Enter the size of the array : ");
    scanf("%d", &size);

    int arr[size];
    printf("Enter %d integers :\n", size);
    for (int i = 0; i < size; i++){
        scanf("%d", &arr[i]);
    }
    printf("Enter the integer to search for : ");
    scanf("%d", &target);

    int i;
    for (i = 0; i < size && arr[i] != target; i++);

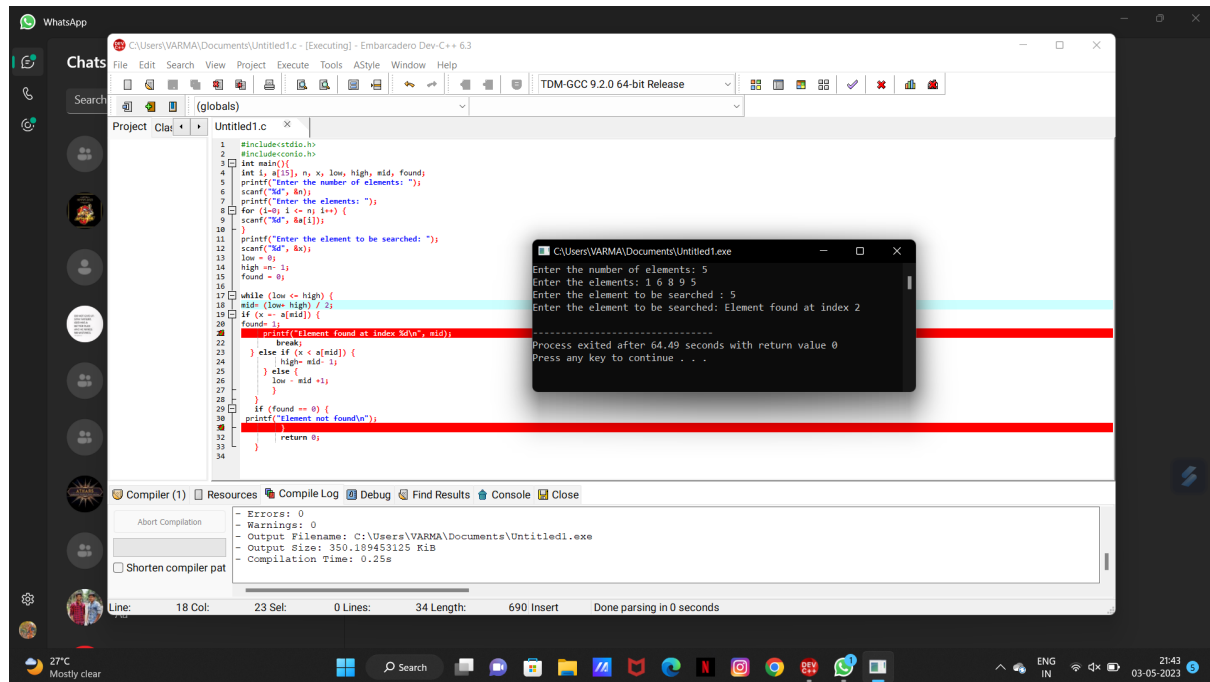
    if (i < size){
        printf("%d found at index %d.\n", target, i);
    } else {
        printf("%d not found in the array.\n", target);
    }

    return 0;
}
```

Enter the size of the array : 5  
Enter 5 integers :  
1 2 3 4 5  
Enter the integer to search for : 3  
3 found at index 2.  
-----  
Process exited after 9.307 seconds with return value 0  
Press any key to continue . . .

Compilation results...  
- Errors: 0  
- Warnings: 0  
- Output Filename: C:\Users\VARMA\Documents\Untitled6.exe  
- Output Size: 128.6015625 KiB  
- Compilation Time: 0.27s

## 19. BINARY SEARCH

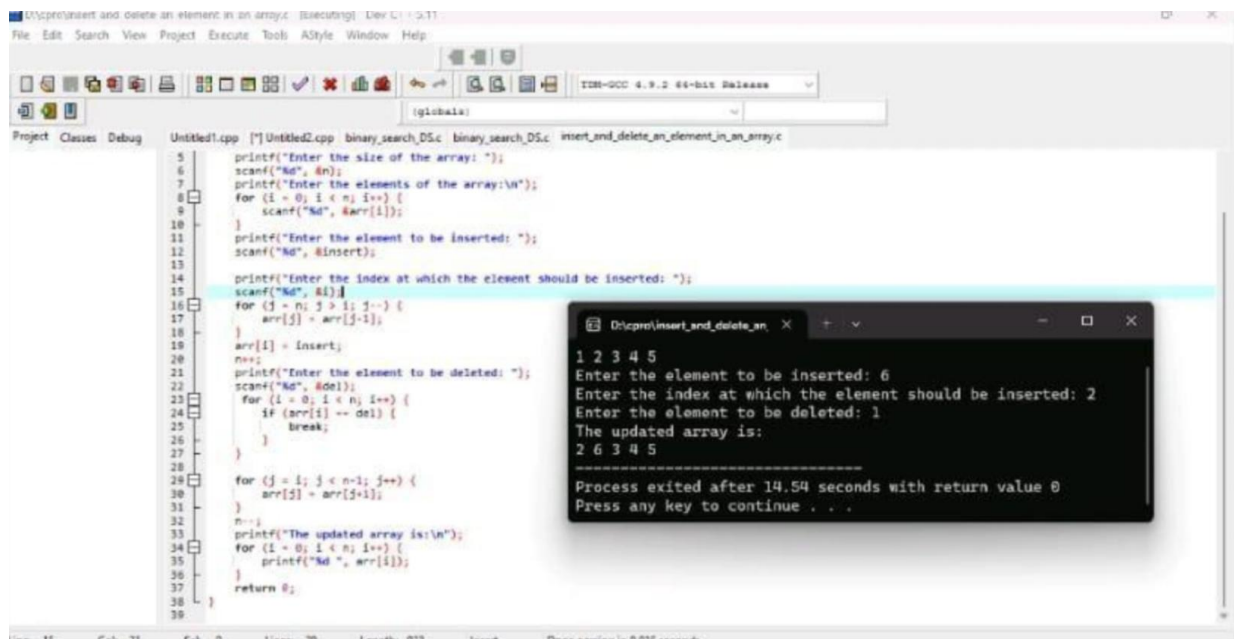


The screenshot shows a C++ IDE with a project named 'Untitled1.c'. The code implements a binary search algorithm. The output window shows the following input and output:

```
Enter the number of elements: 5
Enter the elements: 1 6 8 9 5
Enter the element to be searched: 5
Enter the element to be searched: Element found at index 2
-----
Process exited after 64.49 seconds with return value 0
Press any key to continue . . .
```

```
1 #include <stdio.h>
2 #include <conio.h>
3 int main()
4 {
5     int i, a[5], n, x, low, high, mid, found;
6     printf("Enter the number of elements: ");
7     scanf("%d", &n);
8     for (i=0; i < n; i++) {
9         printf("Enter the elements: ");
10        scanf("%d", &a[i]);
11    }
12    printf("Enter the element to be searched: ");
13    scanf("%d", &x);
14    low = 0;
15    high = n-1;
16    found = 0;
17    while (low <= high) {
18        mid = (low+high) / 2;
19        if (x == a[mid]) {
20            printf("Element found at index %d\n", mid);
21            found = 1;
22            break;
23        } else if (x < a[mid]) {
24            high = mid - 1;
25        } else {
26            low = mid + 1;
27        }
28    }
29    if (found == 0) {
30        printf("Element not found\n");
31        return 0;
32    }
33 }
```

## 20. INSERT AND DELETE A ELEMENT IN AN ARRAY



The screenshot shows a C++ IDE with a project named 'Untitled2.cpp'. The code implements a program to insert and delete an element in an array. The output window shows the following input and output:

```
1 2 3 4 5
Enter the element to be inserted: 6
Enter the index at which the element should be inserted: 2
Enter the element to be deleted: 1
The updated array is:
2 6 3 4 5
-----
Process exited after 14.54 seconds with return value 0
Press any key to continue . . .
```

```
1 #include <stdio.h>
2 #include <conio.h>
3 int main()
4 {
5     printf("Enter the size of the array: ");
6     scanf("%d", &n);
7     printf("Enter the elements of the array:\n");
8     for (i = 0; i < n; i++) {
9         scanf("%d", &arr[i]);
10    }
11    printf("Enter the element to be inserted: ");
12    scanf("%d", &insert);
13    printf("Enter the index at which the element should be inserted: ");
14    scanf("%d", &i);
15    for (j = n; j > i; j--) {
16        arr[j] = arr[j-1];
17    }
18    arr[i] = insert;
19    n++;
20    printf("Enter the element to be deleted: ");
21    scanf("%d", &del);
22    for (i = 0; i < n; i++) {
23        if (arr[i] == del) {
24            break;
25        }
26    }
27    for (j = i; j < n-1; j++) {
28        arr[j] = arr[j+1];
29    }
30    n--;
31    printf("The updated array is:\n");
32    for (i = 0; i < n; i++) {
33        printf("%d ", arr[i]);
34    }
35    return 0;
36 }
```

## 21. INITIALIZATION AND PRINTING OF 2-D ARRAY

The screenshot shows the Dev-C++ IDE with a C++ program for initializing and printing a 2D array. The code is as follows:

```
1 #include<stdio.h>
2 #include<conio.h>
3
4 int main()
5 {
6     int a[10][10],i,j,rows,columns;
7     printf("Enter the number of rows : ");
8     scanf("%d",&rows);
9     printf("Enter the number of columns : ");
10    scanf("%d",&columns);
11
12    for(i=0;i<rows;i++){
13        for(j=0;j<columns;j++){
14            scanf("%d",&a[i][j]);
15            printf("%d",a[i][j]);
16        }
17        printf("\n");
18    }
19 }
20
```

A terminal window is open, showing the program's execution:

```
C:\Users\VARMA\Documents\Untitled9.cpp - [Executing] - Dev-C++ 5.11
Enter the number of rows : 2
Enter the number of columns : 3
1 2 3 4
1 2 3
4
```

The bottom panel shows the compilation results:

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\VARMA\Documents\Untitled9.exe
- Output Size: 128.7734375 KiB
- Compilation Time: 0.25s
```