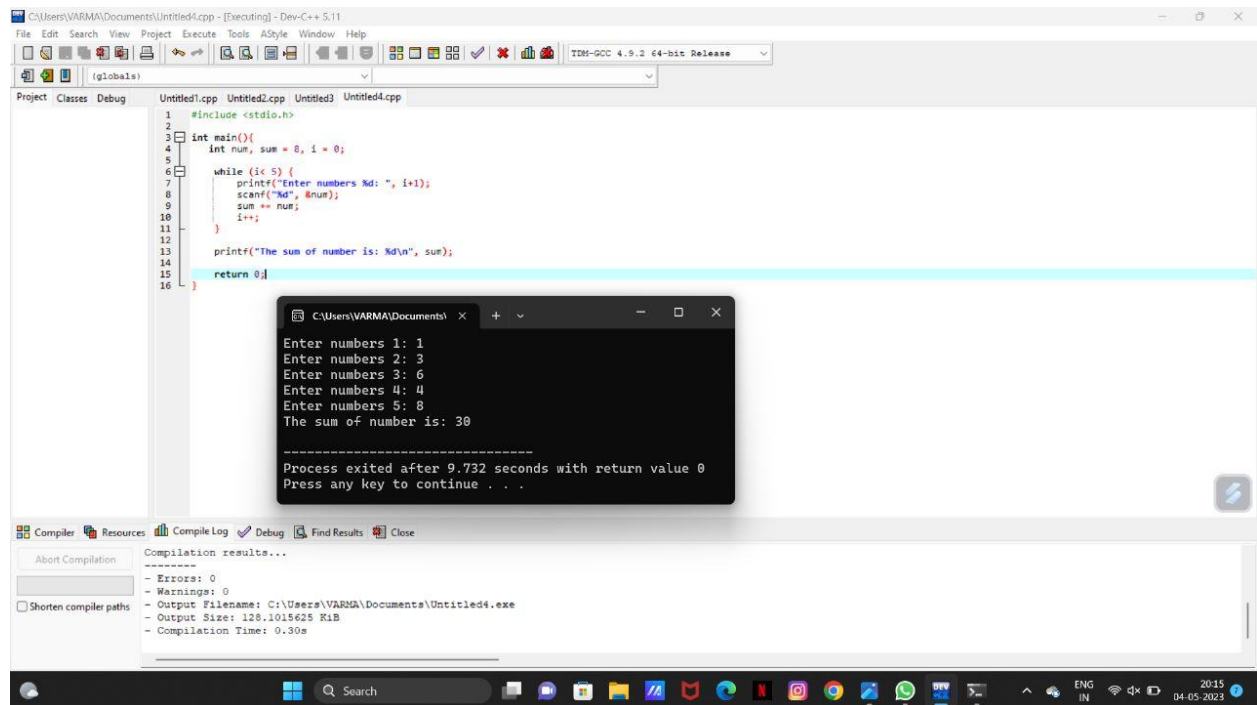


11.GET 5 NUMBERS FROM USER AND SUM IT WHILE LOOP



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

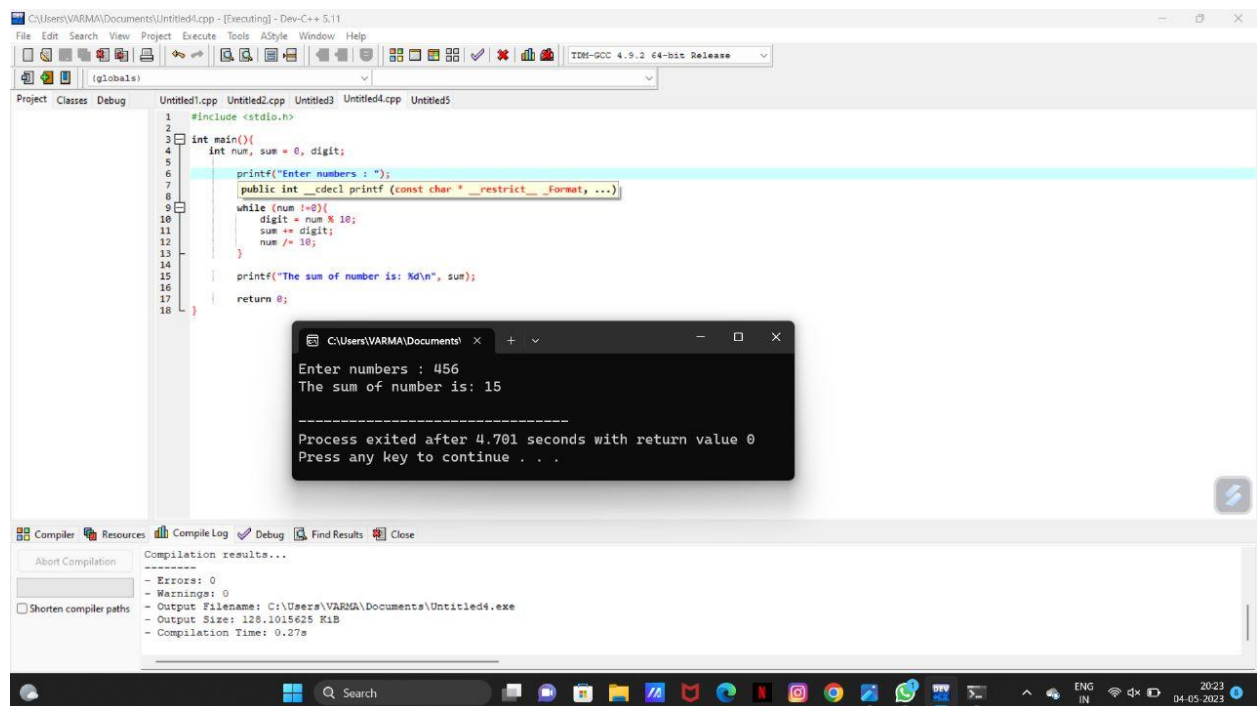
Enter numbers 1: 1
Enter numbers 2: 3
Enter numbers 3: 6
Enter numbers 4: 4
Enter numbers 5: 8
The sum of number is: 30

Process exited after 9.732 seconds with return value 0
Press any key to continue . . .

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\VARMA\Documents\Untitled4.exe
- Output Size: 128.1015625 KiB
- Compilation Time: 0.30s

12. SUM OF DIGITS USING WHILE LOOP



```
1 #include <stdio.h>
2
3 int main(){
4     int num, sum = 0, digit;
5
6     printf("Enter numbers : ");
7     public int __cdecl printf(const char * __restrict __Format, ...)
8
9     while (num != 0){
10        digit = num % 10;
11        sum += digit;
12        num /= 10;
13    }
14
15    printf("The sum of number is: %d\n", sum);
16
17    return 0;
18 }
```

Enter numbers : 456
The sum of number is: 15

Process exited after 4.701 seconds with return value 0
Press any key to continue . . .

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\VARMA\Documents\Untitled4.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 }
```

Console Output:

```
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. INITIALIZATION OF ARRAY

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

Console Output:

```
C:\Users\VARMA\Documents\
Enter the element : 1
1 2 3 4 5 6 7 8 9 10
-----
Process exited after 1.902 seconds with return value 0
Press any key to continue . . .
```

Compiler Output:

```
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\VARMA\Documents\Untitled4.exe
- Output Size: 128.1015625 KiB
- Compilation Time: 0.28s
```

15. SUM OF ARRAY ELEMENTS

The screenshot shows the Dev-C++ IDE with a C++ program open in the editor. The program is titled 'Untitled7.cpp' and is located at 'C:\Users\VARMA\Documents\Untitled7.cpp'. 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 program is executed, and the output is displayed in a separate window titled 'C:\Users\VARMA\Documents\Untitled7.exe'. The output shows the user entering 10 integers (1, 2, 3, 4, 5) and the program calculating the sum as 15.

```
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 . . .
```

16. MINIMUM ELEMENT IN ARRAY

The screenshot shows the Dev-C++ IDE with a C++ program open in the editor. The program is titled 'Untitled7.cpp' and is located at 'C:\Users\VARMA\Documents\Untitled7.cpp'. 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);

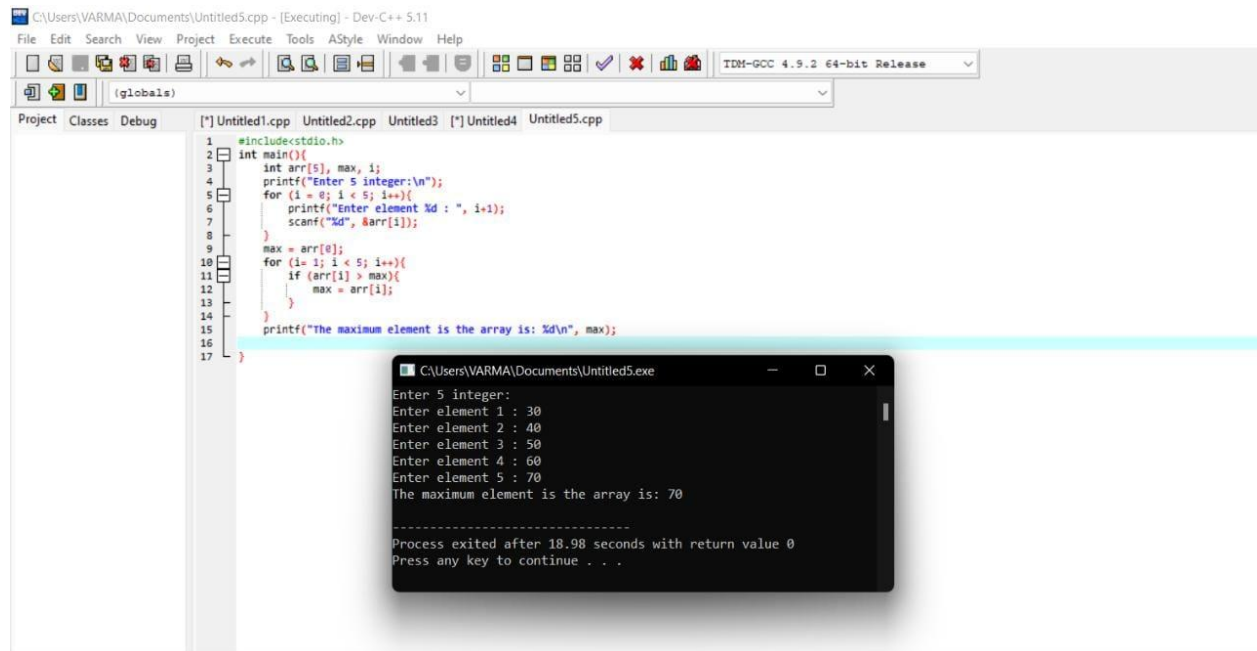
    return 0;
}
```

The program is executed, and the output is displayed in a separate window titled 'C:\Users\VARMA\Documents\Untitled7.exe'. The output shows the user entering 5 integers (10, 30, 50, 60, 100) and the program finding the minimum element as 10.

```
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 . . .
```

17. MAXIMUM ELEMENT IN AN ARRAY



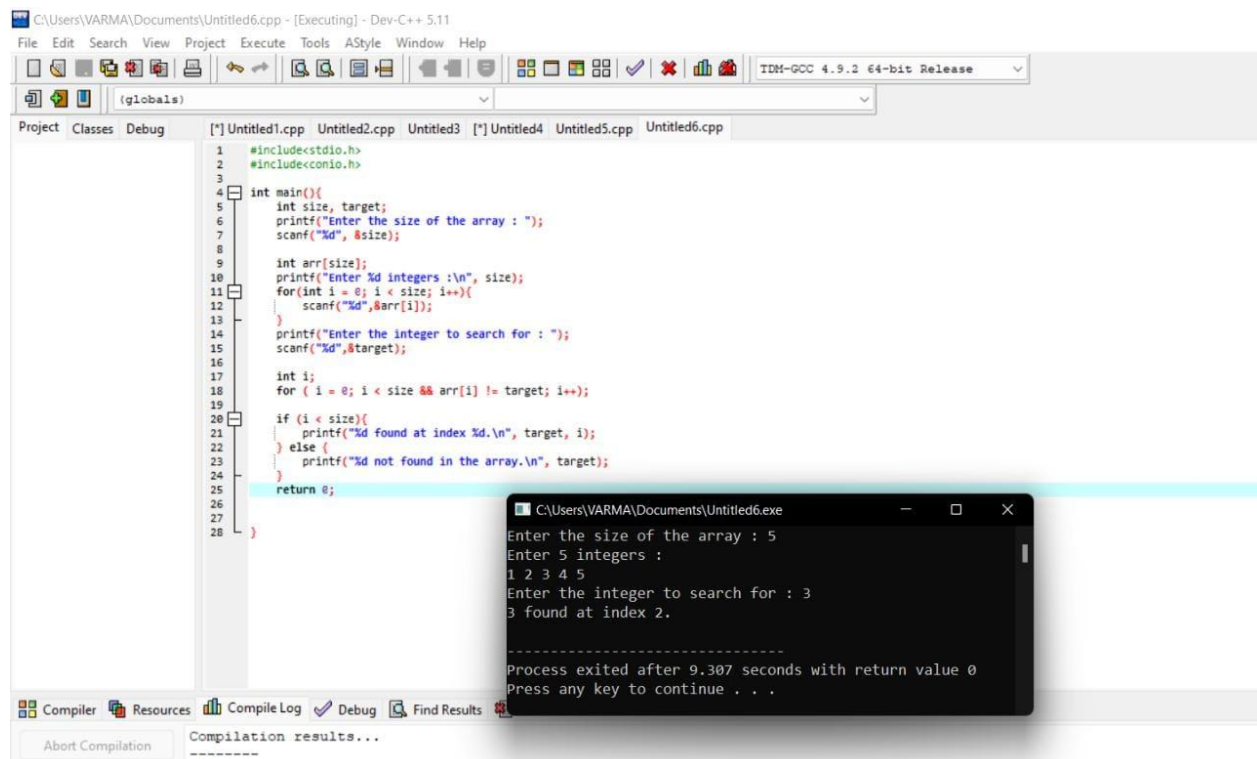
The screenshot shows the Dev-C++ IDE with a C++ program open in 'Untitled5.cpp'. The program includes `<stdio.h>` and defines a `main` function. It declares an array `arr` of size 5, prompts the user to enter 5 integers, and then iterates through the array to find the maximum value. A separate console window titled 'C:\Users\VARMA\Documents\Untitled5.exe' displays the program's output, showing the user entering the values 30, 40, 50, 60, and 70, and the program outputting 'The maximum element is the array is: 70'.

```
1 #include<stdio.h>
2 int main(){
3     int arr[5], max, i;
4     printf("Enter 5 integer:\n");
5     for (i = 0; i < 5; i++){
6         printf("Enter element %d : ", i+1);
7         scanf("%d", &arr[i]);
8     }
9     max = arr[0];
10    for (i= 1; i < 5; i++){
11        if (arr[i] > max){
12            max = arr[i];
13        }
14    }
15    printf("The maximum element is the array is: %d\n", max);
16 }
17 }
```

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 . . .

18. SEARCH ELEMENT IN ARRAY USING LINEAR SEARCH



The screenshot shows the Dev-C++ IDE with a C++ program open in 'Untitled6.cpp'. The program includes `<stdio.h>` and `<conio.h>`. It defines a `main` function that prompts the user for the size of the array and the number of integers to enter. It then prompts for a target integer and uses a linear search loop to find its index. A separate console window titled 'C:\Users\VARMA\Documents\Untitled6.exe' displays the program's output, showing the user entering an array size of 5, the integers 1, 2, 3, 4, 5, and a target of 3, which is found at index 2.

```
1 #include<stdio.h>
2 #include<conio.h>
3
4 int main(){
5     int size, target;
6     printf("Enter the size of the array : ");
7     scanf("%d", &size);
8
9     int arr[size];
10    printf("Enter %d integers :\n", size);
11    for(int i = 0; i < size; i++){
12        scanf("%d", &arr[i]);
13    }
14    printf("Enter the integer to search for : ");
15    scanf("%d", &target);
16
17    int i;
18    for ( i = 0; i < size && arr[i] != target; i++);
19
20    if (i < size){
21        printf("%d found at index %d.\n", target, i);
22    } else {
23        printf("%d not found in the array.\n", target);
24    }
25    return 0;
26 }
27
28 }
```

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 . . .

19. BINARY SEARCH

\\VARMA\Documents\Untitled1.c - [Executing] - Embarcadero Dev-C++ 6.3

Search View Project Execute Tools AStyle Window Help

TDM-GCC 9.2.0 64-bit Release

(globals)

Untitled1.c

```
1 #include<stdio.h>
2 #include<conio.h>
3 int main()
4 {
5     int i, a[15], n, x, low, high, mid, found;
6     printf("Enter the number of elements: ");
7     scanf("%d", &n);
8     printf("Enter the elements: ");
9     for (i=0; i <= n; i++) {
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             found = 1;
21             printf("Element found at index %d\n", mid);
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     }
32     return 0;
33 }
```

C:\Users\VARMA\Documents\Untitled1.exe

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 . . .

er (1) Resources Compile Log Debug Find Results Console Close

Compilation

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\VARMA\Documents\Untitled1.exe
- Output Size: 350.189453125 KiB
- Compilation Time: 0.25s

1 compiler pat

18 Col: 23 Sel: 0 Lines: 34 Length: 690 Insert Done parsing in 0 seconds

20. INSERT AND DELETE A ELEMENT IN AN ARRAY

D:\cpro\insert_and_delete_an_element_in_an_array - [Executing] - Dev-C++ 6.3.1

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Release

(globals)

Project Classes Debug

Untitled1.cpp [?] Untitled2.cpp binary_search_DS.c binary_search_DS.c insert_and_delete_an_element_in_an_array.c

```
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
14 printf("Enter the index at which the element should be inserted: ");
15 scanf("%d", &idx);
16 for (j = n; j > idx; j--) {
17     arr[j] = arr[j-1];
18 }
19 arr[idx] = insert;
20 n++;
21 printf("Enter the element to be deleted: ");
22 scanf("%d", &del);
23 for (i = 0; i < n; i++) {
24     if (arr[i] == del) {
25         break;
26     }
27 }
28 for (j = i; j < n-1; j++) {
29     arr[j] = arr[j+1];
30 }
31 n--;
32 printf("The updated array is:\n");
33 for (i = 0; i < n; i++) {
34     printf("%d ", arr[i]);
35 }
36 return 0;
37 }
```

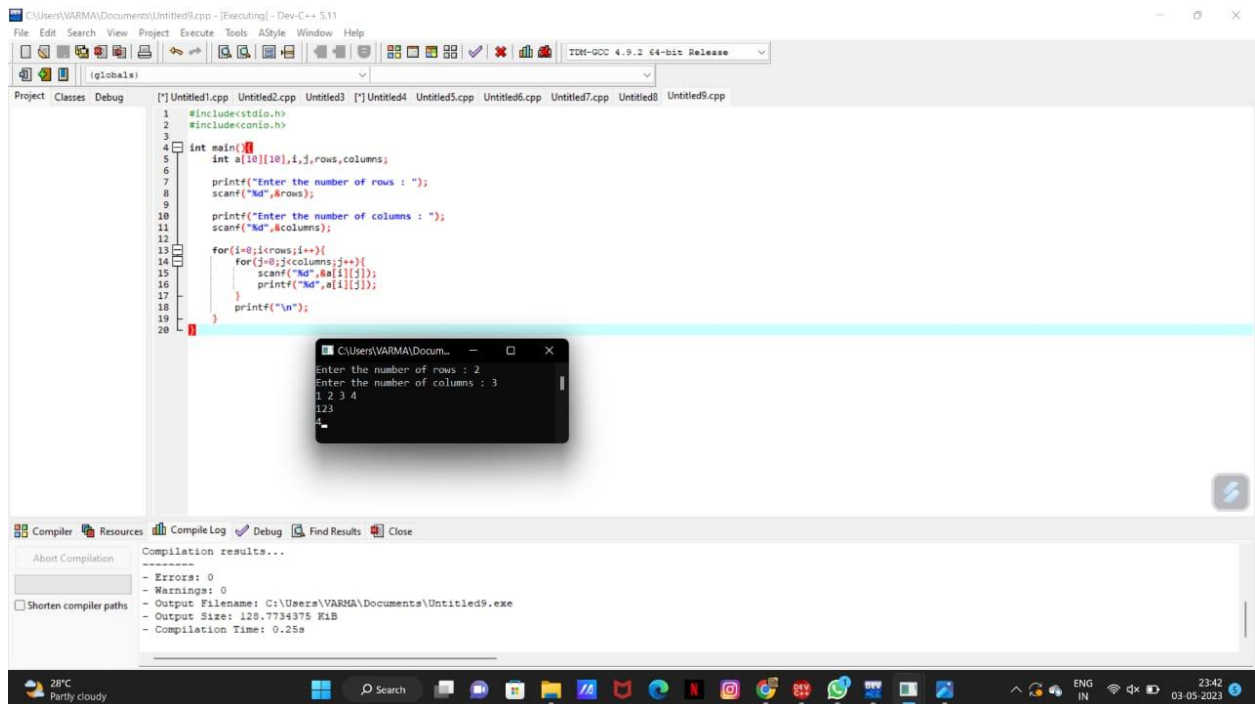
D:\cpro\insert_and_delete_an_

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 . . .

Line: 16 Col: 31 Len: 3 Line: 30 Length: 810 Insert Done parsing in 0.016 seconds

21.INITIALIZATION AND PRINTING OF 2-D ARRAY



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

Enter the number of rows : 2
Enter the number of columns : 3
1 2 3 4
123
4

Compilation results...

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