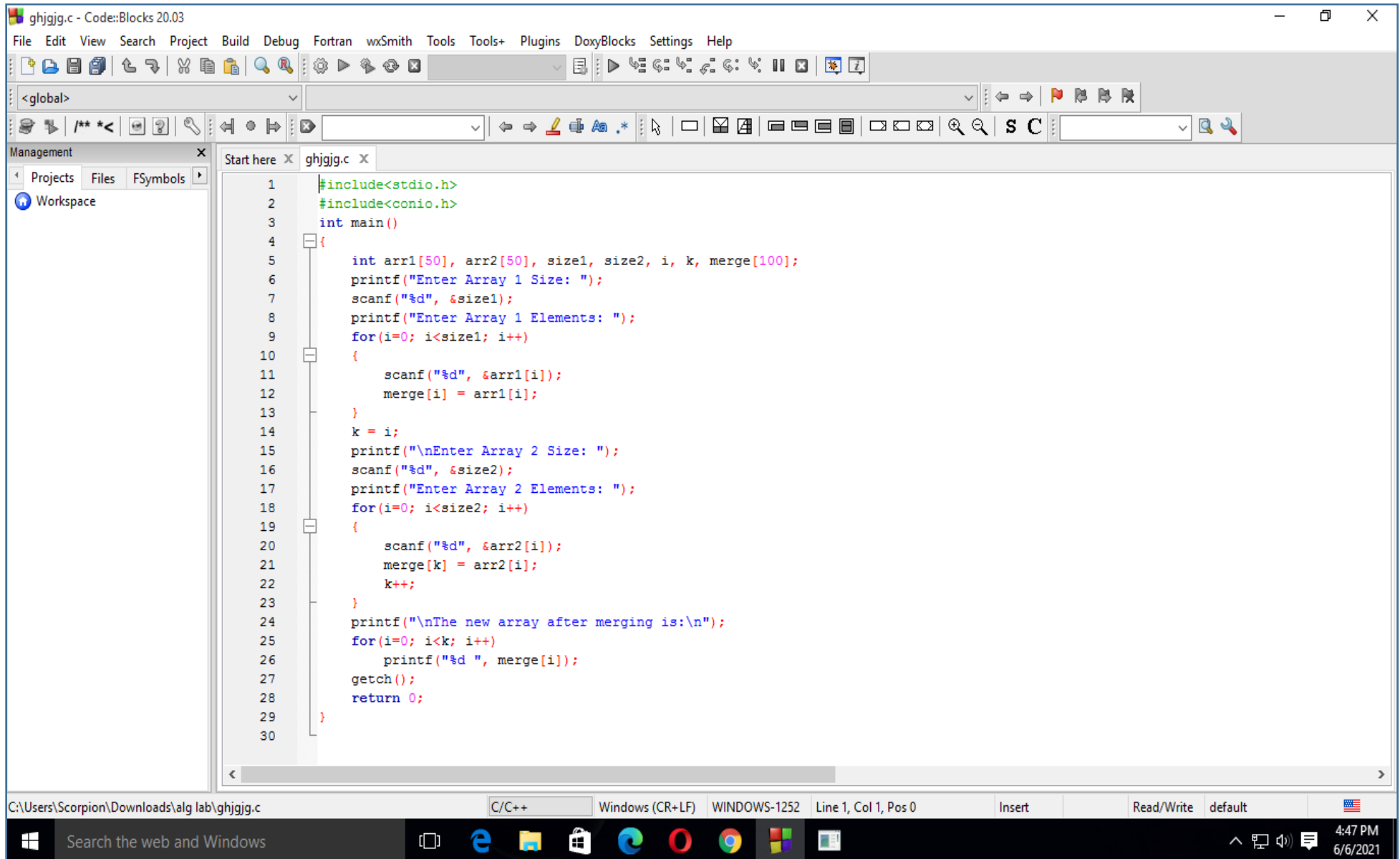
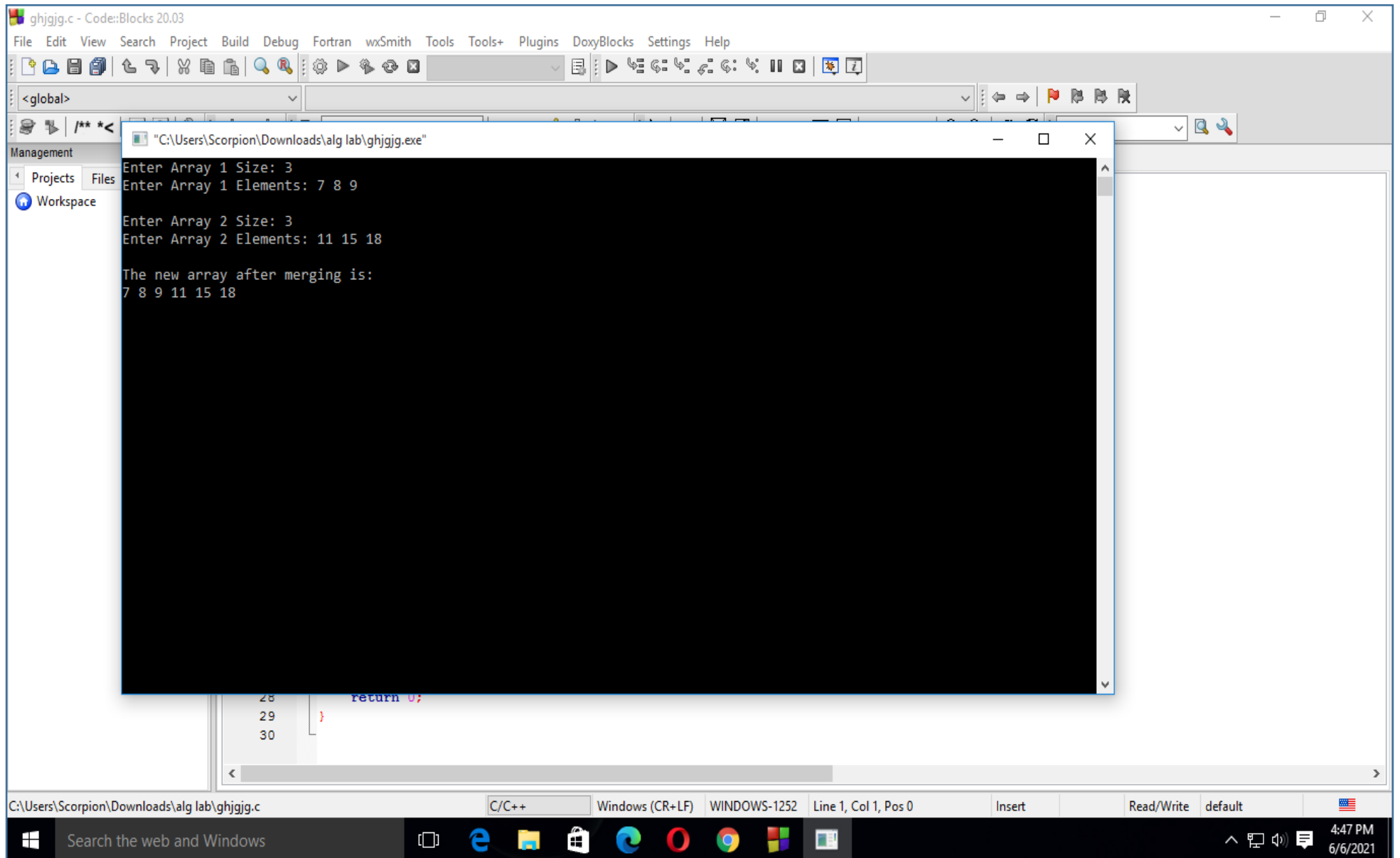


1 no



The screenshot shows the Code::Blocks IDE interface. The title bar reads "ghjgig.c - Code::Blocks 20.03". The menu bar includes File, Edit, View, Search, Project, Build, Debug, Fortran, wxSmith, Tools, Tools+, Plugins, DoxyBlocks, Settings, and Help. The toolbar contains various icons for file operations, editing, and execution. The left sidebar shows the "Management" pane with tabs for Projects, Files, and FSymbols, and a "Workspace" section. The main editor window displays the source code for "ghjgig.c". The code is a C program that merges two arrays into a third. It includes `<stdio.h>` and `<conio.h>`. The `main` function starts by declaring arrays `arr1`, `arr2`, and `merge` with sizes 50, 50, and 100 respectively. It prompts the user to enter the size and elements of the first array, then the second array, and finally prints the merged array. The status bar at the bottom shows the file path "C:\Users\Scorpion\Downloads\alg lab\ghjgig.c", the compiler "C/C++", the encoding "Windows (CR+LF)", the character set "WINDOWS-1252", the cursor position "Line 1, Col 1, Pos 0", and the keyboard layout "Insert". The taskbar at the very bottom shows the Windows Start button, a search bar, and several open applications, with the system clock indicating 4:47 PM on 6/6/2021.

```
1  #include<stdio.h>
2  #include<conio.h>
3  int main()
4  {
5      int arr1[50], arr2[50], size1, size2, i, k, merge[100];
6      printf("Enter Array 1 Size: ");
7      scanf("%d", &size1);
8      printf("Enter Array 1 Elements: ");
9      for(i=0; i<size1; i++)
10     {
11         scanf("%d", &arr1[i]);
12         merge[i] = arr1[i];
13     }
14     k = i;
15     printf("\nEnter Array 2 Size: ");
16     scanf("%d", &size2);
17     printf("Enter Array 2 Elements: ");
18     for(i=0; i<size2; i++)
19     {
20         scanf("%d", &arr2[i]);
21         merge[k] = arr2[i];
22         k++;
23     }
24     printf("\nThe new array after merging is:\n");
25     for(i=0; i<k; i++)
26         printf("%d ", merge[i]);
27     getch();
28     return 0;
29 }
30
```



2 no

The screenshot shows a web browser window with the OnlineGDB website. The browser has three tabs: (2) Facebook, Drunk N High (Official Video) Me, and Online C Compiler - online editor. The address bar shows the URL onlinegdb.com/online_c_compiler. The website's left sidebar contains the OnlineGDB logo, a description "online compiler and debugger for c/c++", and a list of links: code, compile, run, debug, share, IDE, My Projects, Classroom (marked as new), Learn Programming, Programming Questions, Sign Up, and Login. Below these links are social media icons for Facebook and Twitter, and a red button with a plus sign and "55.3K". The main content area displays a C program in a dark-themed editor. The program is a binary search algorithm. It starts with a loop to read 'n' integers into an array. Then, it prompts the user to "Enter value to findn" and reads a 'key'. It then performs a binary search, updating 'low' and 'high' pointers and calculating 'mid'. If the key is found, it prints the location and breaks the loop. If not found, it prints a message. The program ends with a return statement. Below the code editor is an "input" section with a text area containing the following text: "Enter number of elementsn 4", "Enter 4 integersn 1 8 9 12", "Enter value to findn 8", and "8 found at location 2.n". At the bottom of the page, there is a footer with links: About, FAQ, Blog, Terms of Use, Contact Us, GDB Tutorial, Credits, and Privacy. The Windows taskbar at the very bottom shows the search bar and several application icons, with the system clock indicating 5:06 PM on 6/6/2021.

OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.

IDE
My Projects
Classroom **new**
Learn Programming
Programming Questions
Sign Up
Login

55.3K

```
main.c
8  for(i = 0; i < n; i++)
9  scanf("%d",&array[i]);
10 printf("Enter value to findn");
11 scanf("%d", &key);
12 low = 0;
13 high = n - 1;
14 mid = (low+high)/2;
15 while (low <= high) {
16     if(array[mid] < key)
17         low = mid + 1;
18     else if (array[mid] == key) {
19         printf("%d found at location %d.n", key, mid+1);
20         break;
21     }
22     else
23         high = mid - 1;
24     mid = (low + high)/2;
25 }
26 if(low > high)
27     printf("Not found! %d isn't present in the list.n", key);
28 return 0;
29 }
```

input

Enter number of elementsn 4
Enter 4 integersn 1 8 9 12
Enter value to findn 8
8 found at location 2.n
...Program finished with exit code 0
Press ENTER to exit console.

About • FAQ • Blog • Terms of Use • Contact Us
• GDB Tutorial • Credits • Privacy

https://www.onlinegdb.com/online_c_compiler#tab-stdin

Search the web and Windows

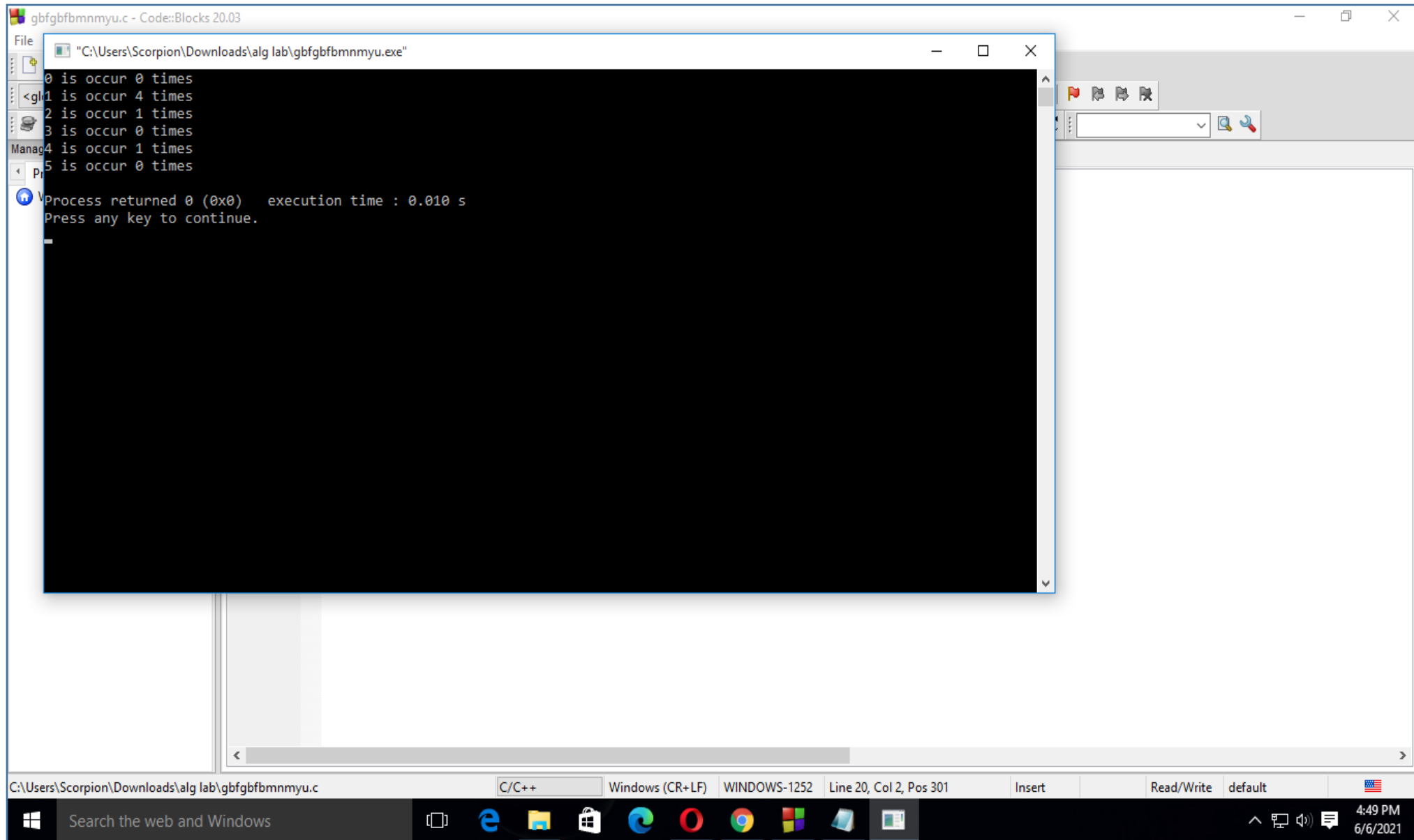
5:06 PM
6/6/2021

3 no

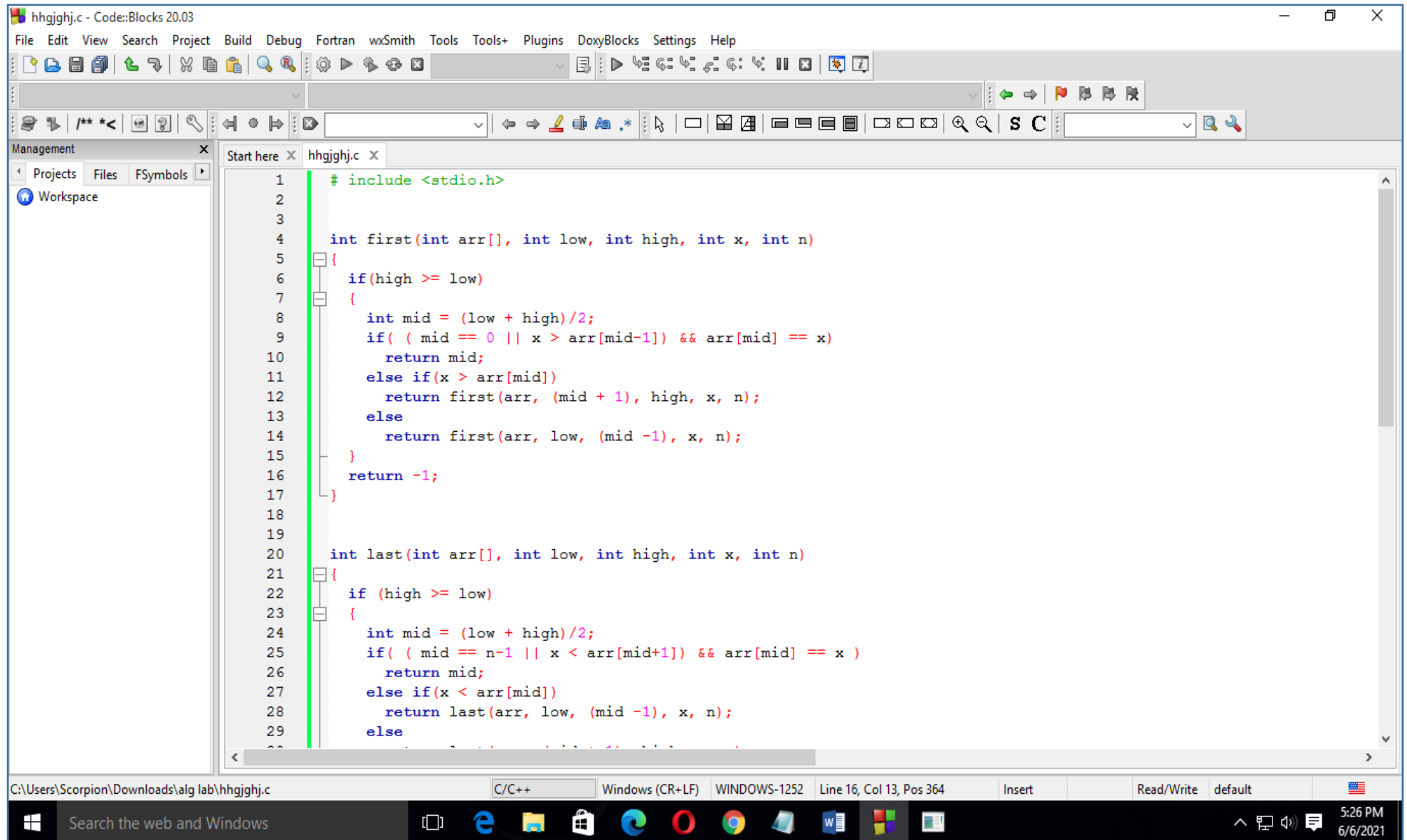
The screenshot shows the Code::Blocks 20.03 IDE with a C program open. The program is designed to count the occurrences of each number in a given array. The code is as follows:

```
1  #include<stdio.h>
2
3  #define SIZE 6
4
5  int main()
6  {
7
8      int A[SIZE]={1,2,1,1,4,1,4};
9
10     int OccurCount[SIZE]={0,0,0,0,0,0};
11
12     for(int i=0; i<SIZE; i++)
13         OccurCount[A[i]]++;
14     for(int i=0; i<SIZE; i++)
15         printf("%d is occur %d times\n", i, OccurCount[i]);
16
17     return 0;
18 }
```

The IDE interface includes a menu bar (File, Edit, View, Search, Project, Build, Debug, Fortran, wxSmith, Tools, Tools+, Plugins, DoxyBlocks, Settings, Help), a toolbar, and a sidebar with tabs for Projects, Files, and FSymbols. The main editor window displays the code, and the status bar at the bottom shows the file path, compiler (C/C++), window title (Windows (CR+LF)), line and column information (Line 20, Col 2, Pos 301), and the current mode (Insert).



4 no



```
1  # include <stdio.h>
2
3
4  int first(int arr[], int low, int high, int x, int n)
5  {
6      if(high >= low)
7      {
8          int mid = (low + high)/2;
9          if( ( mid == 0 || x > arr[mid-1]) && arr[mid] == x)
10             return mid;
11          else if(x > arr[mid])
12             return first(arr, (mid + 1), high, x, n);
13          else
14             return first(arr, low, (mid -1), x, n);
15      }
16      return -1;
17  }
18
19
20 int last(int arr[], int low, int high, int x, int n)
21 {
22     if (high >= low)
23     {
24         int mid = (low + high)/2;
25         if( ( mid == n-1 || x < arr[mid+1]) && arr[mid] == x )
26             return mid;
27         else if(x < arr[mid])
28             return last(arr, low, (mid -1), x, n);
29         else
```

C:\Users\Scorpion\Downloads\alg lab\hhgjghj.c C/C++ Windows (CR+LF) WINDOWS-1252 Line 16, Col 13, Pos 364 Insert Read/Write default 5:26 PM 6/6/2021

