**Object Oriented Programming**

**Lab Journal - Lab # 2**

Name: Saad Ahmad

Enrollment #: 01-134222-130

Class: 2A

**Objective**

This is lab will cover Computer programming concepts including arrays, pointers and dynamic memory allocation.

**Task 1 :**

Do as directed.

|  |  |
| --- | --- |
| **1.** | **Which variable does ip1 point to at the end of the following code?**  int i, j,\*ip1,\*ip2;  ip1=&i;  ip2=&j;  ip1=ip2;  **Solution:**  **j** |
| **2.** | **What does the following program display?**  int main() {  int value1 = 5, value2 = 15;  int \*p1, \*p2;  p1=&value1;  \*p1=10;  \*p2=\*p1;  p1=p2;  \*p1=20;  cout<< value1;  cout<< value2;  return 0;  } |
|  | **Output: 1020** |
| **3.** | **What does the following program display?**  int main() {  int n[5]; int \*p;  p=n; \*p=10;  p++; \*p=20;  p=&n[2]; \*p=30;  p=n+3; \*p=40;  p=n; \*(p+4)=50;  for(int m=0;m<5;m++)  cout<<n[m];  return 0;  } |
| **Output: 10203040** | |
| **4.** | **What does the following program display?**  int main() {  int x[3][4], i , j;  for(i = 0; i<3;++i)  { for(j=0;j<4;++j)  {  x[i][j]=3\*i+j;  cout<<x[i][j]<<" ";  }  cout<<endl;  }  cout<<\*(\*(x+2)+1);  } |
| **Output:**  **0 1 2 3**  **3 4 5 6**  **6 7 8 9**  **7** | |
| **5. Given the following 3 pointers.**  Char \*C1;  Short \*S1;  Int \*I1;  Long \*L1;  And given that they point to memory locations 1000, 2000, 3000, and 4000 respectively. what will be the memory locations pointed to by the pointers after execution of the following?  C1++;  ++S1;  --I1;  L1--; | |
| **Output:**  **1001**  **2002**  **2996**  **3996** | |

**Exercise 1**

Declare an array of 10 integers and get user input to fill the array values. Then, find total number of prime numbers in an array.

**Code :**

#include <iostream>

using namespace std;

int prime(int arr[], int size) {

int a = 0;

int check = 0;

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

for (int j = 1; j <= arr[i]; j++) {

if (arr[i] % j == 0) {

a++;

}

}

if (a == 2) {

check++;

}

a = 0;

}

cout << "This array has " << check << " prime numbers" << endl;

return check;

}

int main()

{

const int size = 10;

int arr[size];

cout << "Enter 10 integers : " << endl;

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

cin >> arr[i];

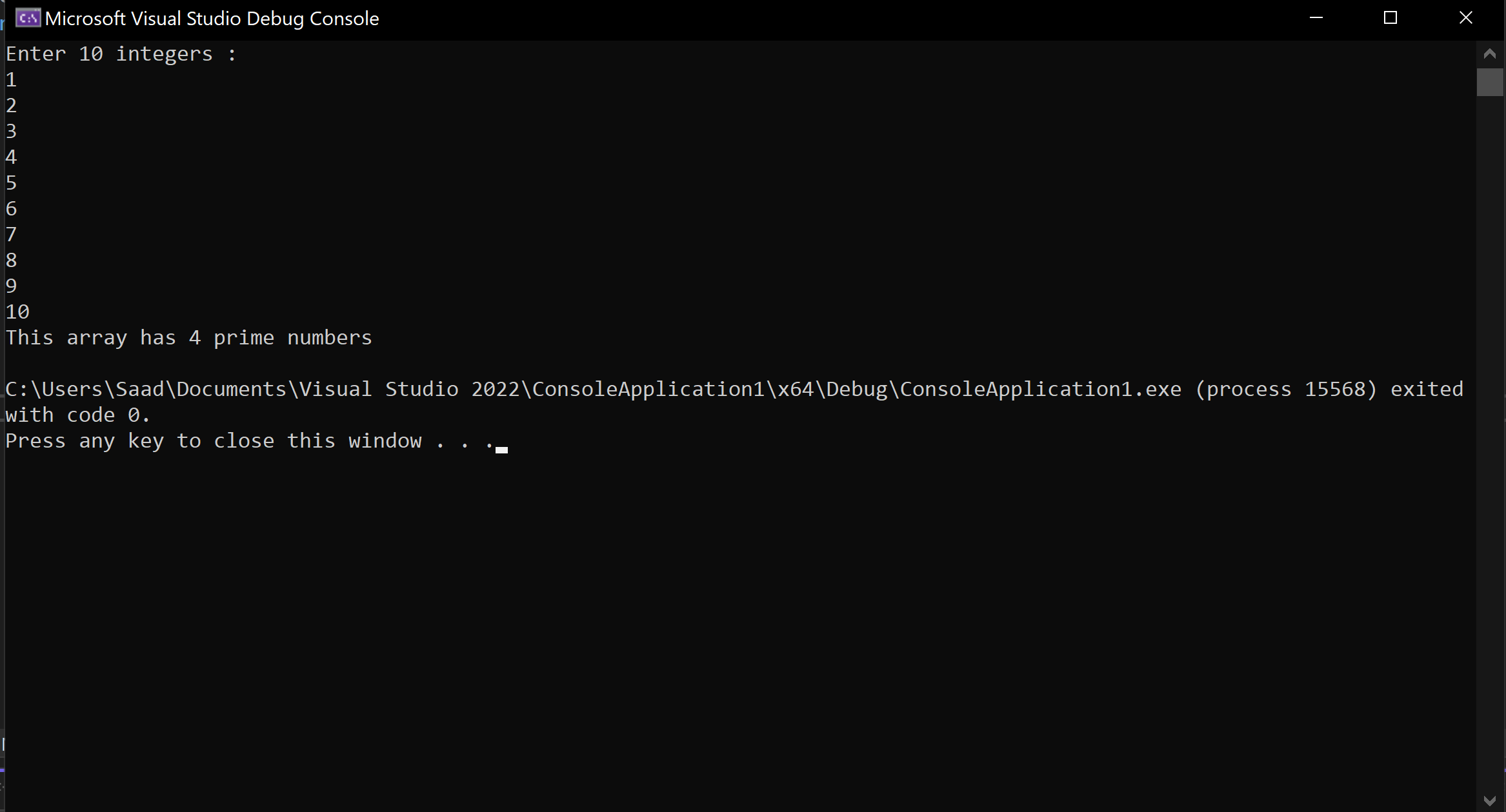
}

prime(arr, size);

return 0;

}

**Output :**

****

**Exercise 2**

Declare an array of size taken by user. Get user input to fill the array values. Then, find the factorial of minimum number in an array.

**Code :**

#include <iostream>

using namespace std;

int factorial(int num) {

int a = num;

int sum = 1;

while (num > 1) {

sum = sum \* num;

num--;

}

cout << "Factorial of " << a << " is "<< sum ;

return 0;

}

int min(int arr[], int size) {

int a = arr[0];

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

if (arr[i] < a) {

a = arr[i];

}

}

factorial(a);

return 0;

}

int main()

{

int size;

cout << "Enter size of the array : " << endl;

cin >> size;

cout << "Enter the elements of the array : " << endl;

int\* arr = new int[size];

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

cin >> arr[i];

}

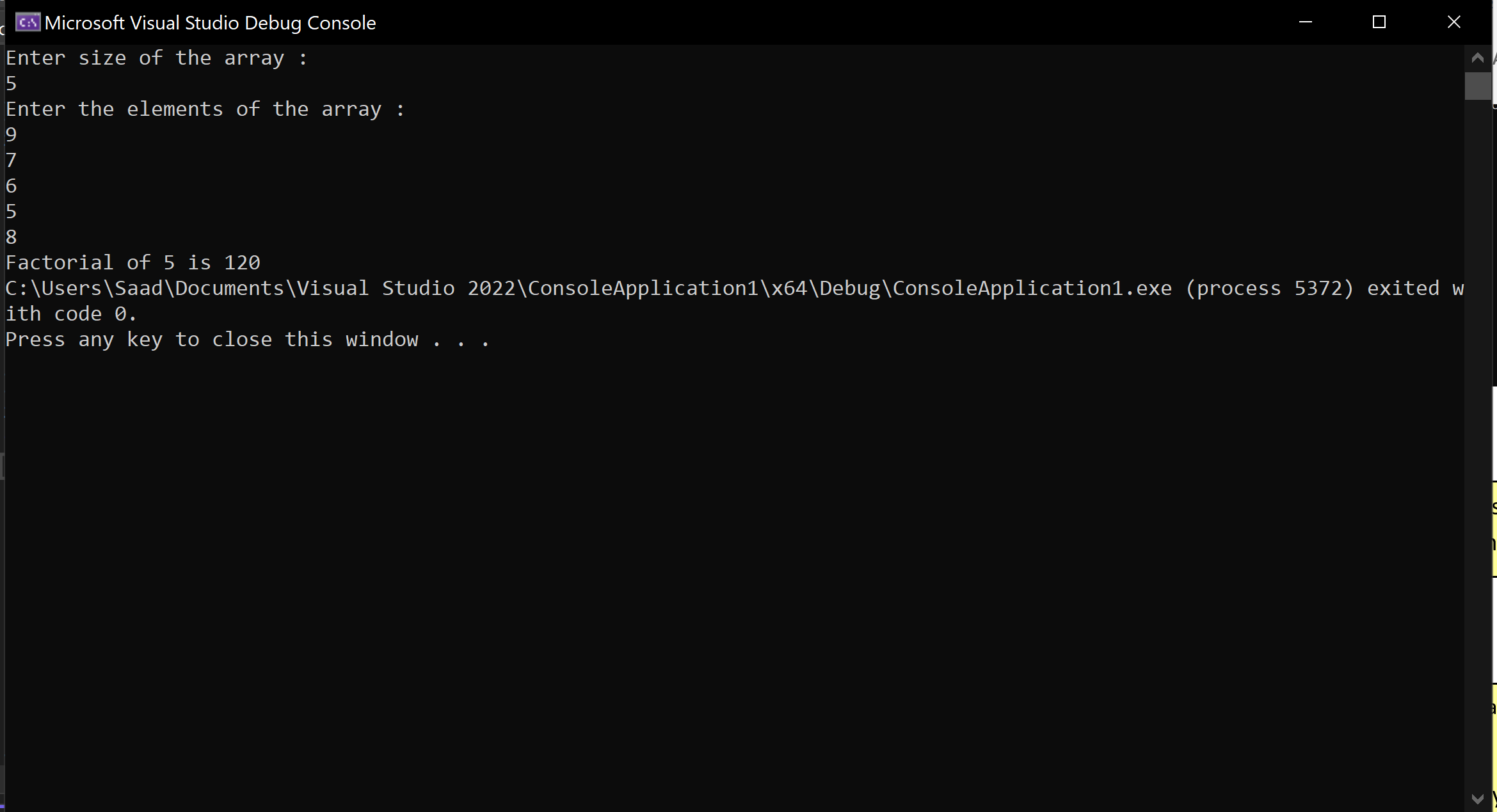
min(arr, size);

delete[] arr;

return 0;

}

**Output :**

****

**Exercise 3**

Write a program to declare an array of user provided size to perform the following functionality.

The program should display the subscript of the cell containing the largest of the values in the array. Thus, for example, if the array looks like this:

Graphical user interface, application, chat or text message

Description automatically generated

then the integer 2 should be displayed as its value. If there is more than one cell containing the largest of the values in the array, then it should print the smallest of the subscripts of the cells containing the largest values.

**Code :**

#include <iostream>

using namespace std;

int max(int arr[], int size) {

int a = arr[0];

int x = 0;

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

if (arr[i] > a) {

a = arr[i];

if (x < i) {

x = i;

}

}

}

return x;

}

int main()

{

int size;

cout << "Enter size of the array : " << endl;

cin >> size;

cout << "Enter the elements of the array : " << endl;

int\* arr = new int[size];

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

cin >> arr[i];

}

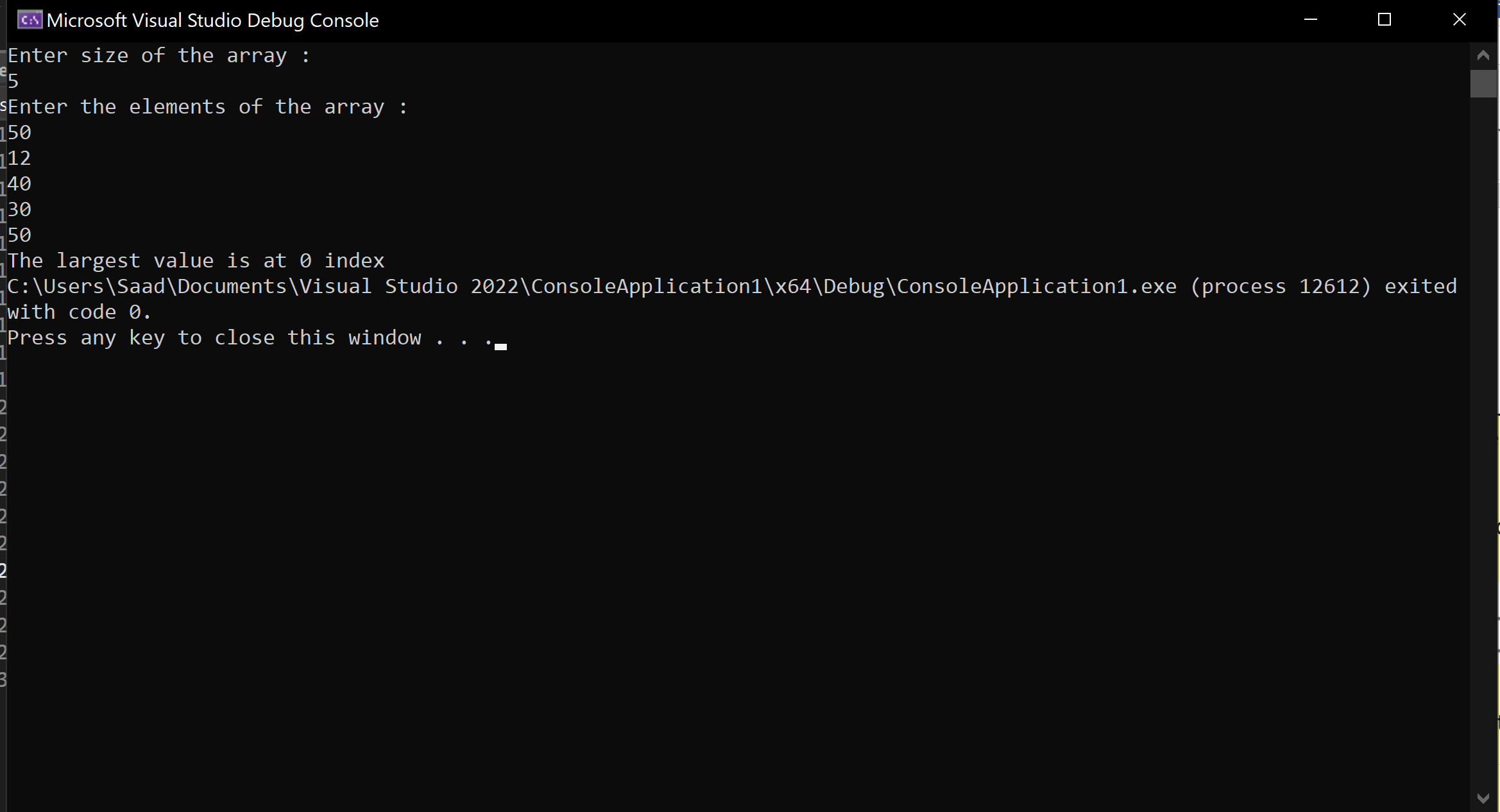
cout << "The largest value is at " << max(arr, size) << " index";

delete[] arr;

return 0;

}

**Output :**

****

**Exercise 4**

Write a program using functions that will initialize an array with N random numbers in the range 1-999 and count the number of integers whose value is **less than the average** value of the integers. Your program is to display the average integer value and the count of integers less than average.

Hint: Divide the program into functions depending on what you want to do i.e. initialize, find average, count numbers less than average value. Main can be used to call the functions.

++++++++++++++++++++++++

**Code:**

#include <iostream>

#include <cstdlib>

using namespace std;

int numCheck(int arr[], int size, int avg) {

int check = 0;

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

if (arr[i] < avg) {

check++;

}

}

cout << "Integers less than the avg value are " << check << endl;

return 0;

}

int avg(int arr[], int size) {

int sum = 0;

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

sum += arr[i];

}

float avg = sum / size;

cout << "Average integer value is " << avg << endl;

numCheck(arr, size, avg);

return 0;

}

int initialize(int arr[], int size) {

srand(time(0));

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

arr[i] = rand() % 1000;

}

avg(arr, size);

return 0;

}

int main()

{

int size;

cout << "Enter size of the array : " << endl;

cin >> size;

int\* arr = new int[size];

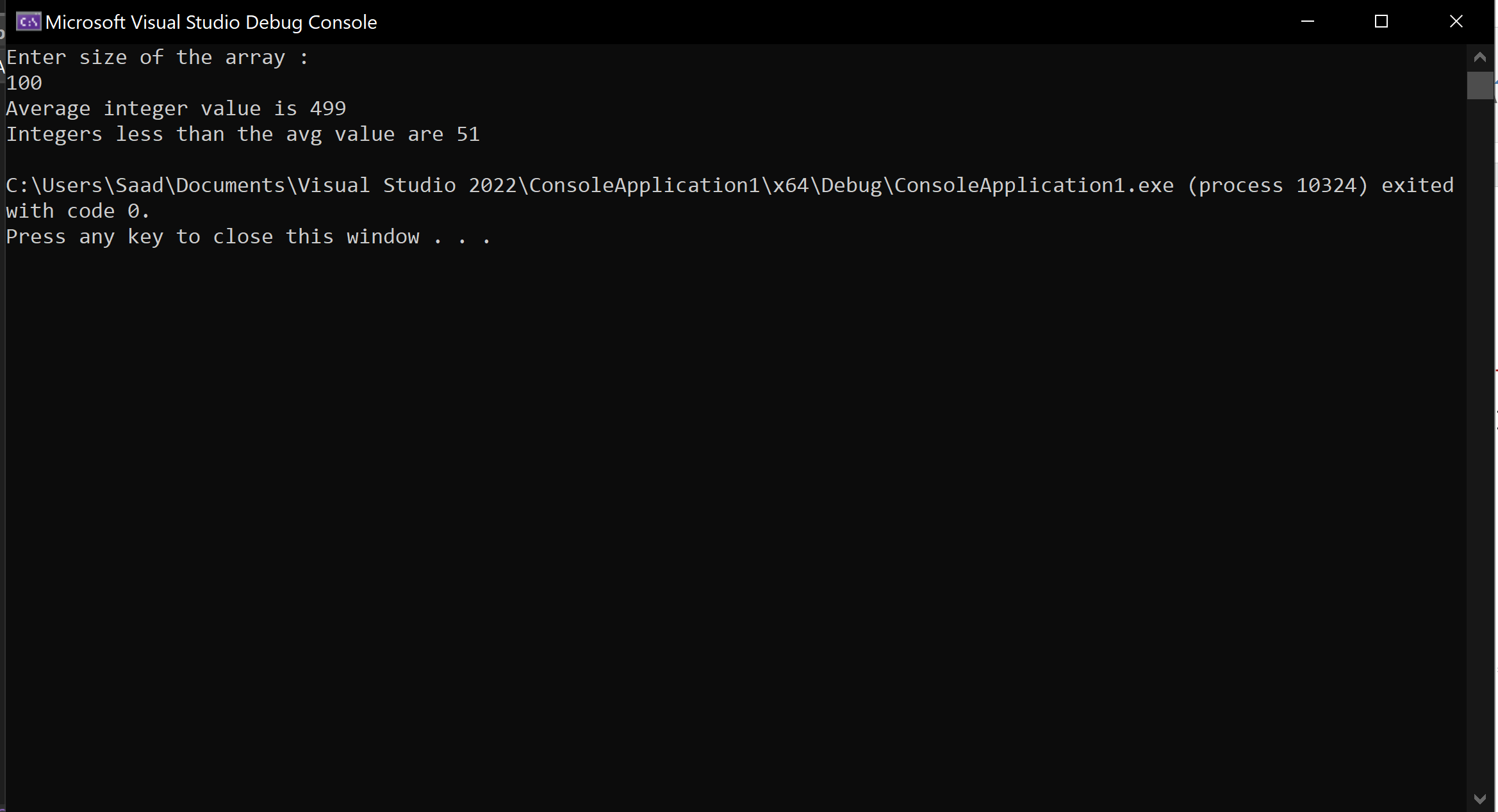
initialize(arr, size);

delete[] arr;

return 0;

}

**Output**:



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*