Comsats University Islamabad, Vehari Campus

Assignment NO:01

Submitted By:

Muhammad Usman Azhar

Submitted To:

Mam Yasmeen Jana

Subject:

Data Structure and Algorithm

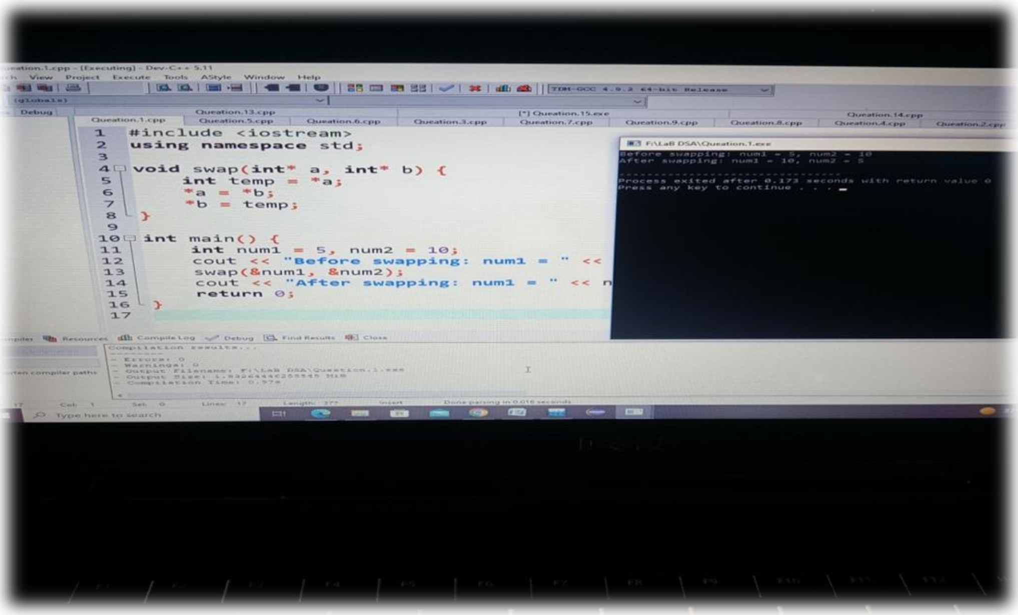
Registration No:

SP22-BCS-092(B)

Date of submission: **10/09/2023**

1. Write a c++ program to swap two numbers by using pointers

```
#include <iostream>
using namespace std;

void swap(int* a, int* b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}

int main() {
    int num1 = 5, num2 = 10;
    cout << "Before swapping: num1 = " << num1 << ", num2 = " << num2 <<
endl;
    swap(&num1, &num2);
    cout << "After swapping: num1 = " << num1 << ", num2 = " << num2 <<
endl;
    return 0;
}
```



```
#include <iostream>
using namespace std;

void swap(int* a, int* b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}

int main() {
    int num1 = 5, num2 = 10;
    cout << "Before swapping: num1 = " <<
    swap(&num1, &num2);
    cout << "After swapping: num1 = " << n
    return 0;
}
```

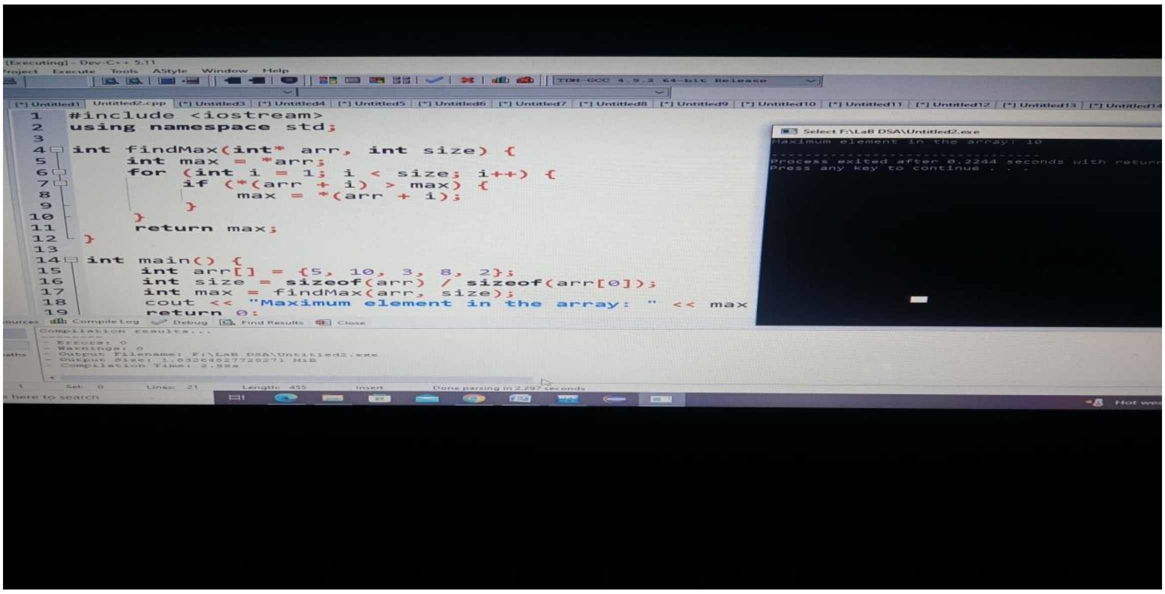
F:\LaB DSA\Queation.1.exe

```
Before swapping: num1 = 5, num2 = 10
After swapping: num1 = 10, num2 = 5

--------------------------------
Process exited after 0.173 seconds with return value 0
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Queation.1.exe
- Output Size: 1.83264446258545 MiB
- Compilation Time: 0.97s
```

2.Write a C++ program to find maximum element in an array by using pointers

```
#include <iostream>
using namespace std;

int findMax(int* arr, int size) {
    int max = *arr;
    for (int i = 1; i < size; i++) {
        if (*(arr + i) > max) {
            max = *(arr + i);
        }
    }
    return max;
}

int main() {
    int arr[] = {5, 10, 3, 8, 2};
    int size = sizeof(arr) / sizeof(arr[0]);
    int max = findMax(arr, size);
    cout << "Maximum element in the array: " << max << endl;
    return 0;
}
```



[Executing] - Dev-C++ 5.11
Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
[*] Untitled1 Untitled2.cpp [*] Untitled3 [*] Untitled4 [*] Untitled5 [*] Untitled6 [*] Untitled7 [*] Untitled8 [*] Untitled9 [*] Untitled10 [*] Untitled11 [*] Untitled12 [*] Untitled13
Compile Log Debug Find Results Close
Select F:\LaB DSA\Untitled2.exe

```
#include <iostream>
using namespace std;

int findMax(int* arr, int size) {
    int max = *arr;
    for (int i = 1; i < size; i++) {
        if (*(arr + i) > max) {
            max = *(arr + i);
        }
    }
    return max;
}

int main() {
    int arr[] = {5, 10, 3, 8, 2};
    int size = sizeof(arr) / sizeof(arr[0]);
    int max = findMax(arr, size);
    cout << "Maximum element in the array: " << max
    return 0;
```

```
Maximum element in the array: 10
--------------------------------
Process exited after 0.2244 seconds with return
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Untitled2.exe
- Output Size: 1.83264827728271 MiB
- Compilation Time: 2.88s
```

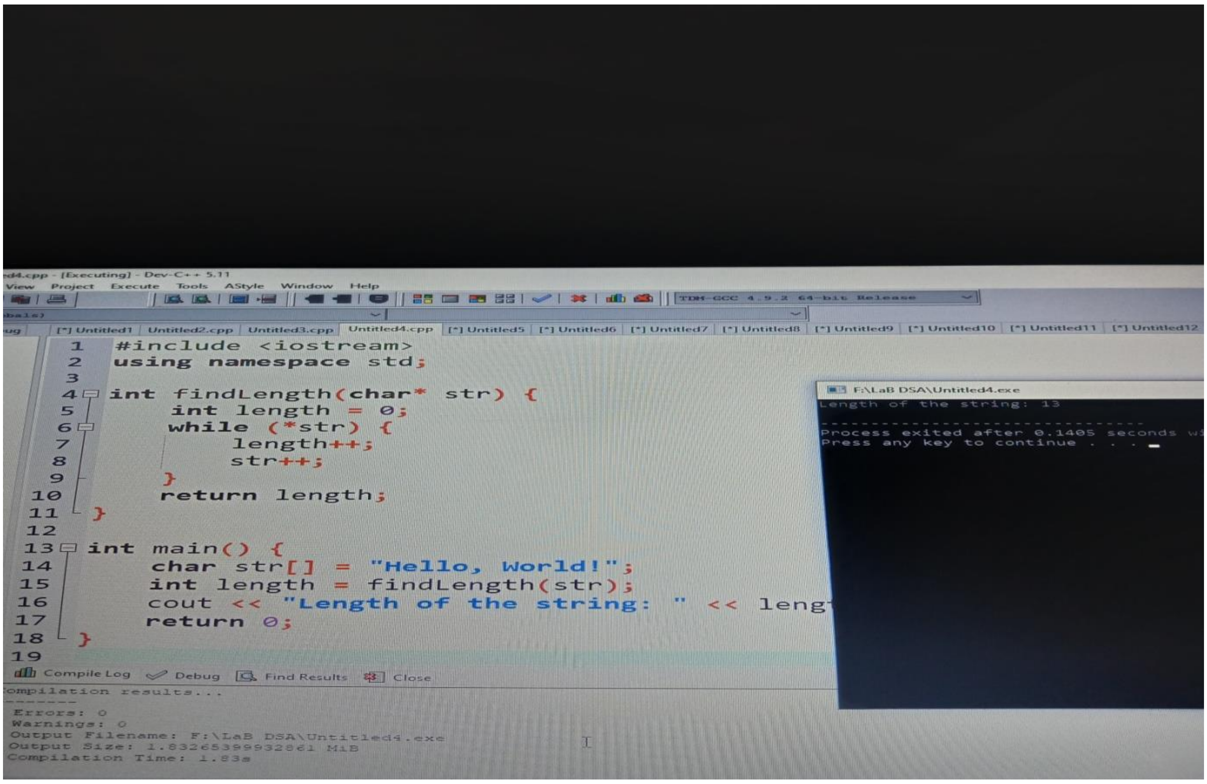
Sel: 0 Lines: 21 Length: 455 Insert Done parsing in 2.297 seconds
here to search
Hot wea

3.Write a C++ program to reverse a string by using pointers

```
#include <iostream>
using namespace std;

void reverseString(char* str) {
    char* start = str;
    char* end = str;
    while (*end) {
        end++;
    }
    end--;

    while (start < end) {
        char temp = *start;
        *start = *end;
        *end = temp;
        start++;
        end--;
    }
}

int main() {
    char str[] = "Hello, World!";
    cout << "Before reversing: " << str << endl;
    reverseString(str);
    cout << "After reversing: " << str << endl;
    return 0;
}
```

4.Write a C++ program to find a length of string by using pointers

```
#include <iostream>
using namespace std;

int findLength(char* str) {
    int length = 0;
    while (*str) {
        length++;
        str++;
    }
    return length;
}

int main() {
    char str[] = "Hello, World!";
    int length = findLength(str);
    cout << "Length of the string: " << length << endl;
    return 0;
}
```



d4.cpp - [Executing] - Dev-C++ 5.11
View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
[*] Untitled1 Untitled2.cpp Untitled3.cpp Untitled4.cpp [*] Untitled5 [*] Untitled6 [*] Untitled7 [*] Untitled8 [*] Untitled9 [*] Untitled10 [*] Untitled11 [*] Untitled12
Compile Log
Debug
Find Results
Close
F:\LaB DSA\Untitled4.exe

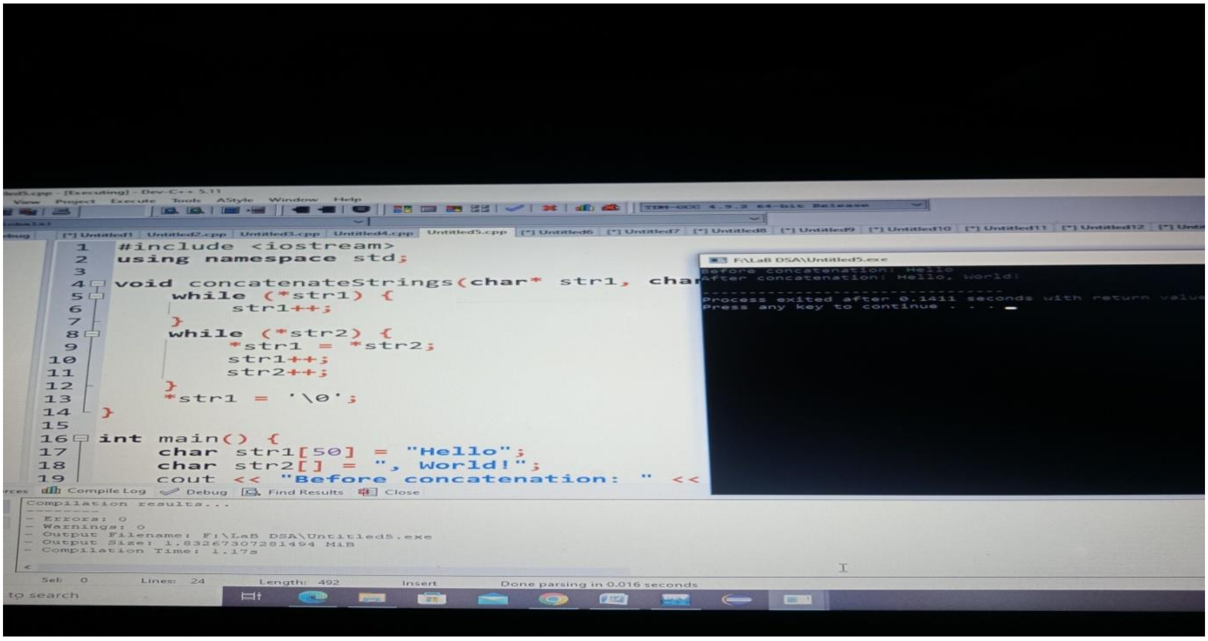
```
#include <iostream>
using namespace std;

int findLength(char* str) {
    int length = 0;
    while (*str) {
        length++;
        str++;
    }
    return length;
}

int main() {
    char str[] = "Hello, World!";
    int length = findLength(str);
    cout << "Length of the string: " << leng
    return 0;
}
```

```
Length of the string: 13

--------------------------------
Process exited after 0.1405 seconds w
Press any key to continue . . .
```

```
ompilation results...
--------
 Errors: 0
 Warnings: 0
 Output Filename: F:\LaB DSA\Untitled4.exe
 Output Size: 1.83265399932861 MiB
 Compilation Time: 1.83s
```

5.Write a C++ program to concatenate two string by using pointers

```
#include <iostream>
using namespace std;

void concatenateStrings(char* str1, char* str2) {
    while (*str1) {
        str1++;
    }
    while (*str2) {
        *str1 = *str2;
        str1++;
        str2++;
    }
    *str1 = '\0';
}

int main() {
    char str1[50] = "Hello";
    char str2[] = ", World!";
    cout << "Before concatenation: " << str1 << endl;
    concatenateStrings(str1, str2);
    cout << "After concatenation: " << str1 << endl;
    return 0;
}
```



```
#include <iostream>
using namespace std;

void concatenateStrings(char* str1, char
    while (*str1) {
        str1++;
    }
    while (*str2) {
        *str1 = *str2;
        str1++;
        str2++;
    }
    *str1 = '\0';
}

int main() {
    char str1[50] = "Hello";
    char str2[] = ", World!";
    cout << "Before concatenation: " <<
```

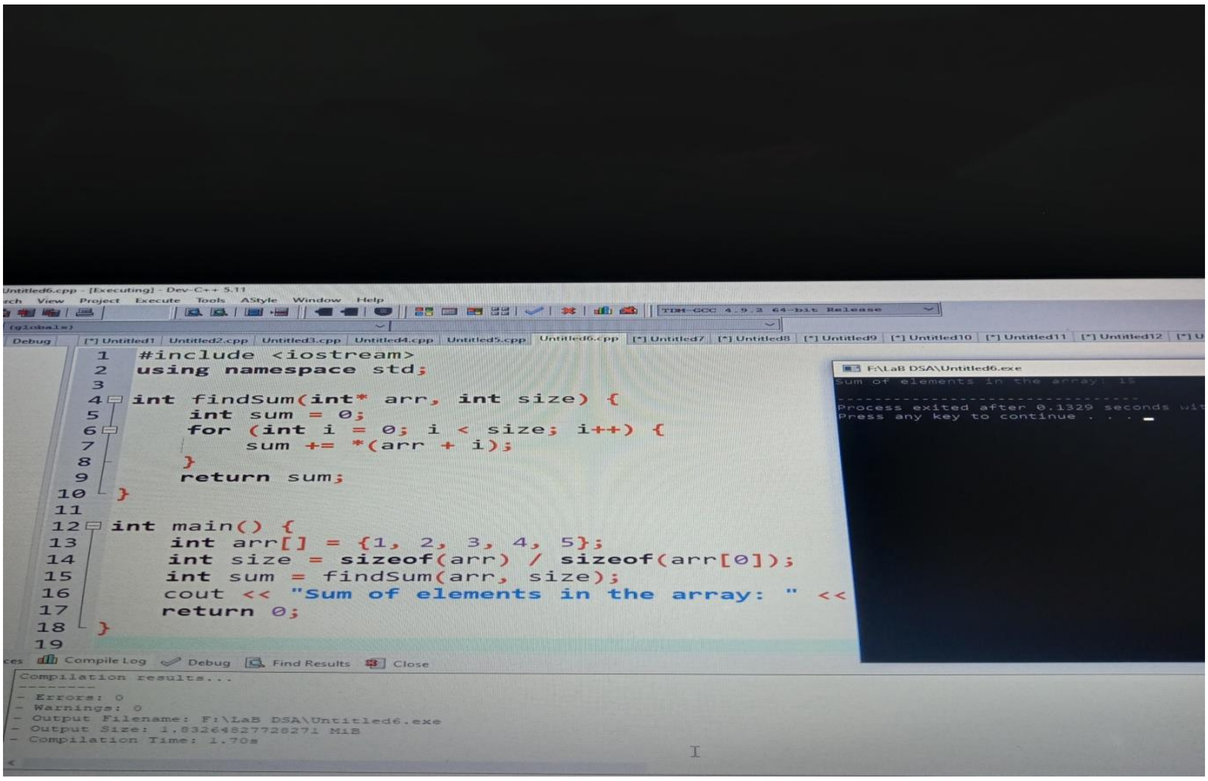
```
Before concatenation: Hello
After concatenation: Hello, World!
--------------------------------
Process exited after 0.1411 seconds with return value
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Untitled5.exe
- Output Size: 1.83267307281494 MiB
- Compilation Time: 1.17s
```

Sel: 0 Lines: 24 Length: 492 Insert Done parsing in 0.016 seconds

6.Write a C++ program to find the sum of elements in an array by using pointers

```
#include <iostream>
using namespace std;

int findSum(int* arr, int size) {
    int sum = 0;
    for (int i = 0; i < size; i++) {
        sum += *(arr + i);
    }
    return sum;
}

int main() {
    int arr[] = {1, 2, 3, 4, 5};
    int size = sizeof(arr) / sizeof(arr[0]);
    int sum = findSum(arr, size);
    cout << "Sum of elements in the array: " << sum << endl;
    return 0;
}
```



Untitled6.cpp - [Executing] - Dev-C++ 5.11
rch View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Debug
[*] Untitled1 Untitled2.cpp Untitled3.cpp Untitled4.cpp Untitled5.cpp Untitled6.cpp [*] Untitled7 [*] Untitled8 [*] Untitled9 [*] Untitled10 [*] Untitled11 [*] Untitled12 [*] U
ces Compile Log Debug Find Results Close

```
#include <iostream>
using namespace std;

int findSum(int* arr, int size) {
    int sum = 0;
    for (int i = 0; i < size; i++) {
        sum += *(arr + i);
    }
    return sum;
}

int main() {
    int arr[] = {1, 2, 3, 4, 5};
    int size = sizeof(arr) / sizeof(arr[0]);
    int sum = findSum(arr, size);
    cout << "Sum of elements in the array: " <<
    return 0;
}
```

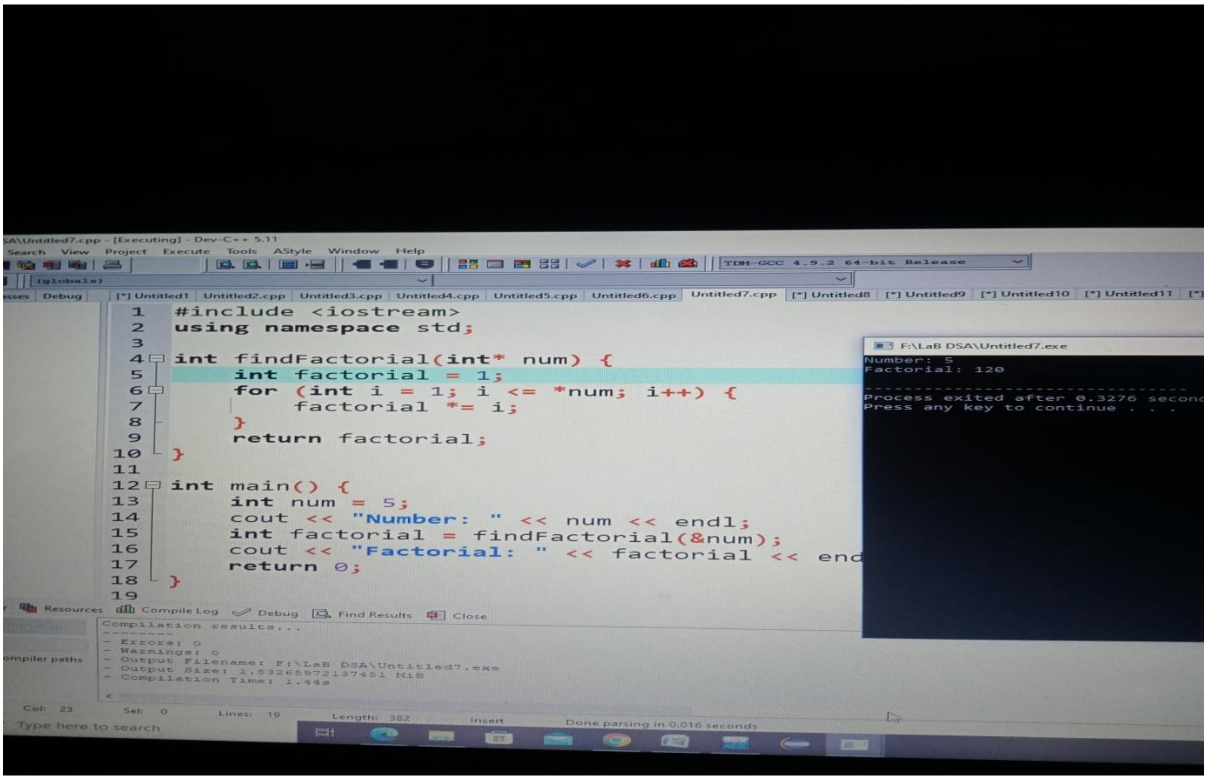
F:\LaB DSA\Untitled6.exe

```
Sum of elements in the array: 15

--------------------------------
Process exited after 0.1329 seconds wit
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Untitled6.exe
- Output Size: 1.83264827728271 MiB
- Compilation Time: 1.70s
```

7.Write a program to find the factorial of number by using pointers

```
#include <iostream>
using namespace std;

int findFactorial(int* num) {
    int factorial = 1;
    for (int i = 1; i <= *num; i++) {
        factorial *= i;
    }
    return factorial;
}

int main() {
    int num = 5;
    cout << "Number: " << num << endl;
    int factorial = findFactorial(&num);
    cout << "Factorial: " << factorial << endl;
    return 0;
}
```



SA\Untitled7.cpp - [Executing] - Dev-C++ 5.11
Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Debug
[*] Untitled1 Untitled2.cpp Untitled3.cpp Untitled4.cpp Untitled5.cpp Untitled6.cpp Untitled7.cpp [*] Untitled8 [*] Untitled9 [*] Untitled10 [*] Untitled11
F:\LaB DSA\Untitled7.exe
Resources Compile Log Debug Find Results Close
ompiler paths
Col: 23 Sel: 0 Lines: 19 Length: 382 Insert Done parsing in 0.016 seconds
Type here to search

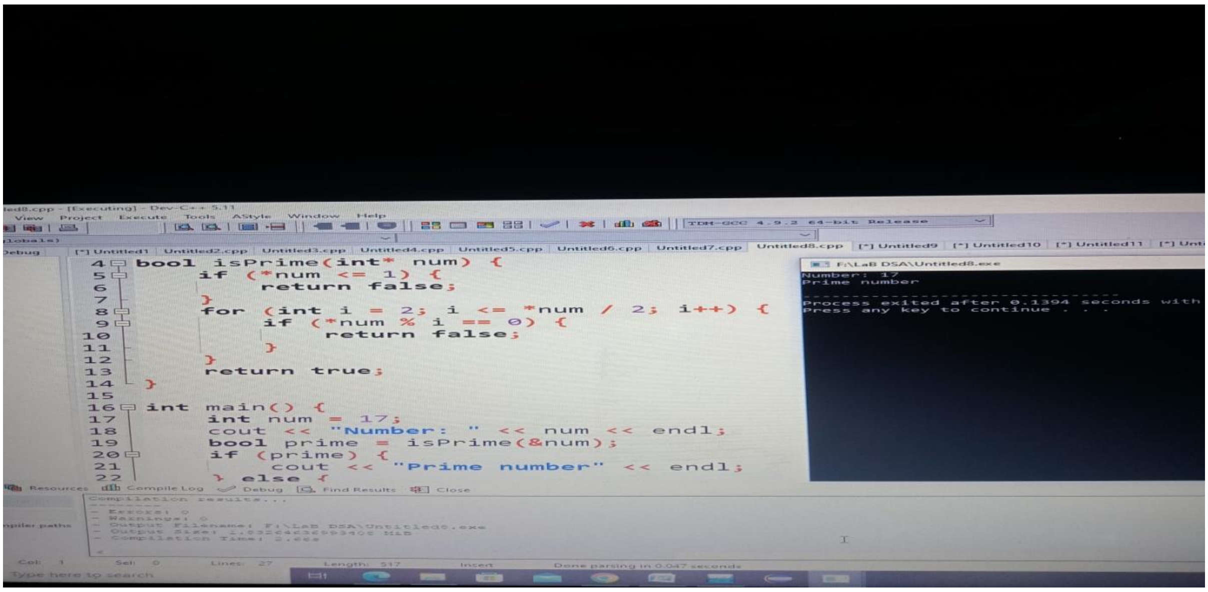
```
#include <iostream>
using namespace std;

int findFactorial(int* num) {
    int factorial = 1;
    for (int i = 1; i <= *num; i++) {
        factorial *= i;
    }
    return factorial;
}

int main() {
    int num = 5;
    cout << "Number: " << num << endl;
    int factorial = findFactorial(&num);
    cout << "Factorial: " << factorial << end
    return 0;
}
```

```
Number: 5
Factorial: 120

--------------------------------
Process exited after 0.3276 secon
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Untitled7.exe
- Output Size: 1.83265972137451 MiB
- Compilation Time: 1.44s
```

8.Write a program to check if a number is prime by using pointers

```
#include <iostream>
using namespace std;

bool isPrime(int* num) {
    if (*num <= 1) {
        return false;
    }
    for (int i = 2; i <= *num / 2; i++) {
        if (*num % i == 0) {
            return false;
        }
    }
    return true;
}

int main() {
    int num = 17;
    cout << "Number: " << num << endl;
    bool prime = isPrime(&num);
    if (prime) {
        cout << "Prime number" << endl;
    } else {
        cout << "Not a prime number" << endl;
    }
    return 0;
}
```



led8.cpp - [Executing] - Dev-C++ 5.11
View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
[*] Untitled1 Untitled2.cpp Untitled3.cpp Untitled4.cpp Untitled5.cpp Untitled6.cpp Untitled7.cpp Untitled8.cpp [*] Untitled9 [*] Untitled10 [*] Untitled11
F:\LaB DSA\Untitled8.exe
Resources Compile Log Debug Find Results Close
Col: 1 Sel: 0 Lines: 27 Length: 517 Insert Done parsing in 0.047 seconds
Type here to search

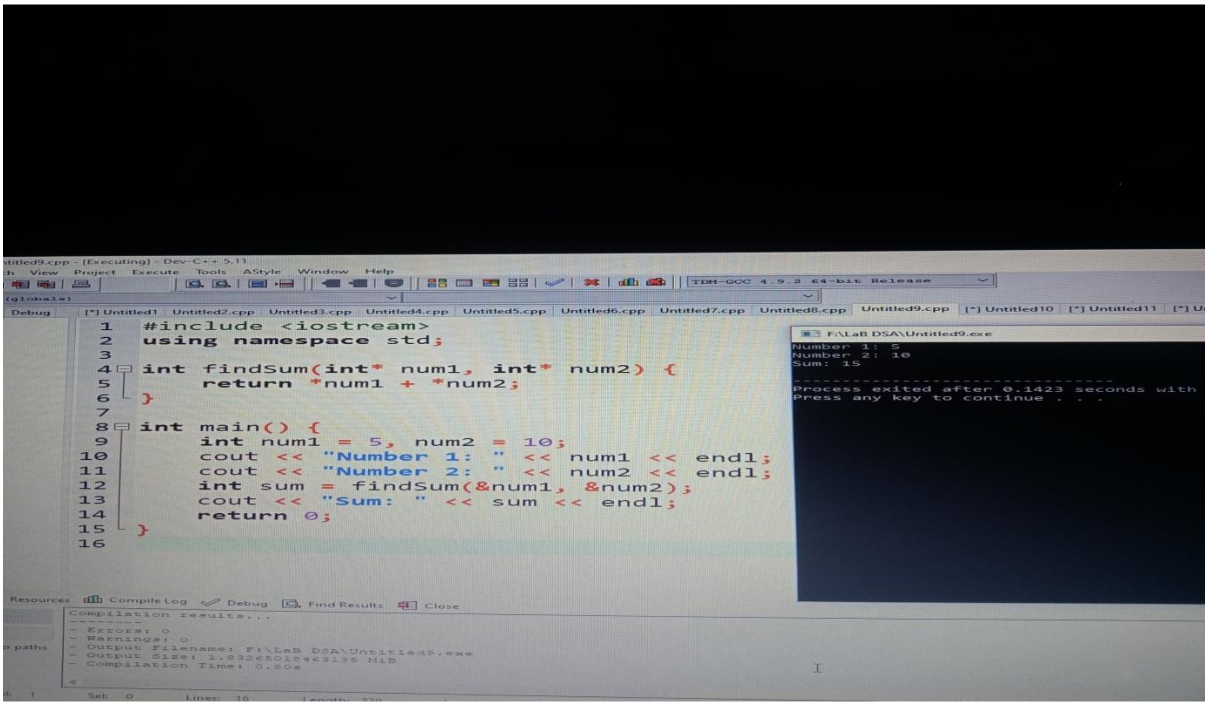
```
bool isPrime(int* num) {
    if (*num <= 1) {
        return false;
    }
    for (int i = 2; i <= *num / 2; i++) {
        if (*num % i == 0) {
            return false;
        }
    }
    return true;
}

int main() {
    int num = 17;
    cout << "Number: " << num << endl;
    bool prime = isPrime(&num);
    if (prime) {
        cout << "Prime number" << endl;
    } else {
```

```
Number: 17
Prime number

--------------------------------
Process exited after 0.1394 seconds with
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Untitled8.exe
- Output Size: 1.83264636993408 MiB
- Compilation Time: 2.66s
```

9.Write a program to find a sum of two numbers by using pointers

```
#include <iostream>
using namespace std;

int findSum(int* num1, int* num2) {
    return *num1 + *num2;
}

int main() {
    int num1 = 5, num2 = 10;
    cout << "Number 1: " << num1 << endl;
    cout << "Number 2: " << num2 << endl;
    int sum = findSum(&num1, &num2);
    cout << "Sum: " << sum << endl;
    return 0;
}
```



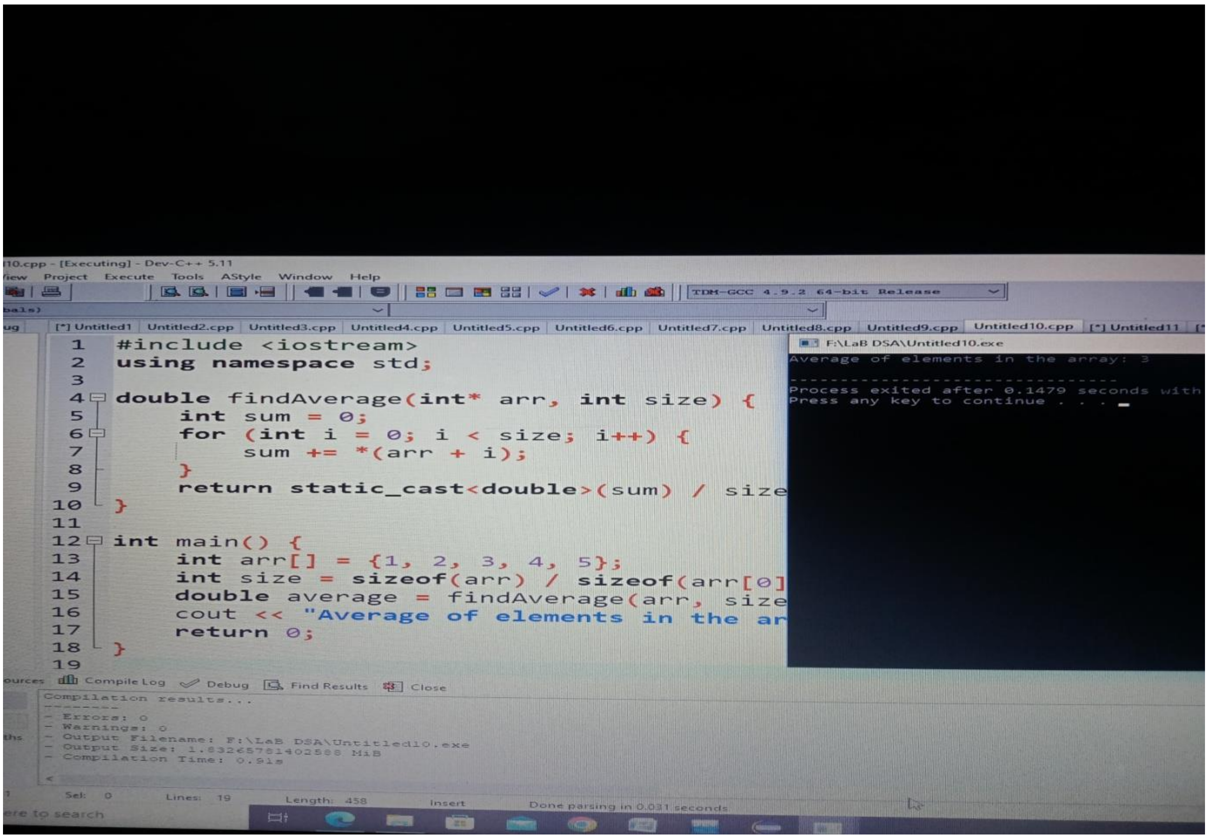
```
#include <iostream>
using namespace std;

int findSum(int* num1, int* num2) {
    return *num1 + *num2;
}

int main() {
    int num1 = 5, num2 = 10;
    cout << "Number 1: " << num1 << endl;
    cout << "Number 2: " << num2 << endl;
    int sum = findSum(&num1, &num2);
    cout << "Sum: " << sum << endl;
    return 0;
}
```

```
Number 1: 5
Number 2: 10
Sum: 15

--------------------------------
Process exited after 0.1423 seconds with
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Untitled9.exe
- Output Size: 1.83265018463135 MiB
- Compilation Time: 0.80s
```

10.Write a C++ program to find a average element in an array by using pointers

```
#include <iostream>
using namespace std;

double findAverage(int* arr, int size) {
    int sum = 0;
    for (int i = 0; i < size; i++) {
        sum += *(arr + i);
    }
    return static_cast<double>(sum) / size;
}

int main() {
    int arr[] = {1, 2, 3, 4, 5};
    int size = sizeof(arr) / sizeof(arr[0]);
    double average = findAverage(arr, size);
    cout << "Average of elements in the array: " << average << endl;
    return 0;
}
```



10.cpp - [Executing] - Dev-C++ 5.11
View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
[*] Untitled1 Untitled2.cpp Untitled3.cpp Untitled4.cpp Untitled5.cpp Untitled6.cpp Untitled7.cpp Untitled8.cpp Untitled9.cpp Untitled10.cpp [*] Untitled11
F:\LaB DSA\Untitled10.exe
Compile Log Debug Find Results Close
Sel: 0 Lines: 19 Length: 458 Insert Done parsing in 0.031 seconds

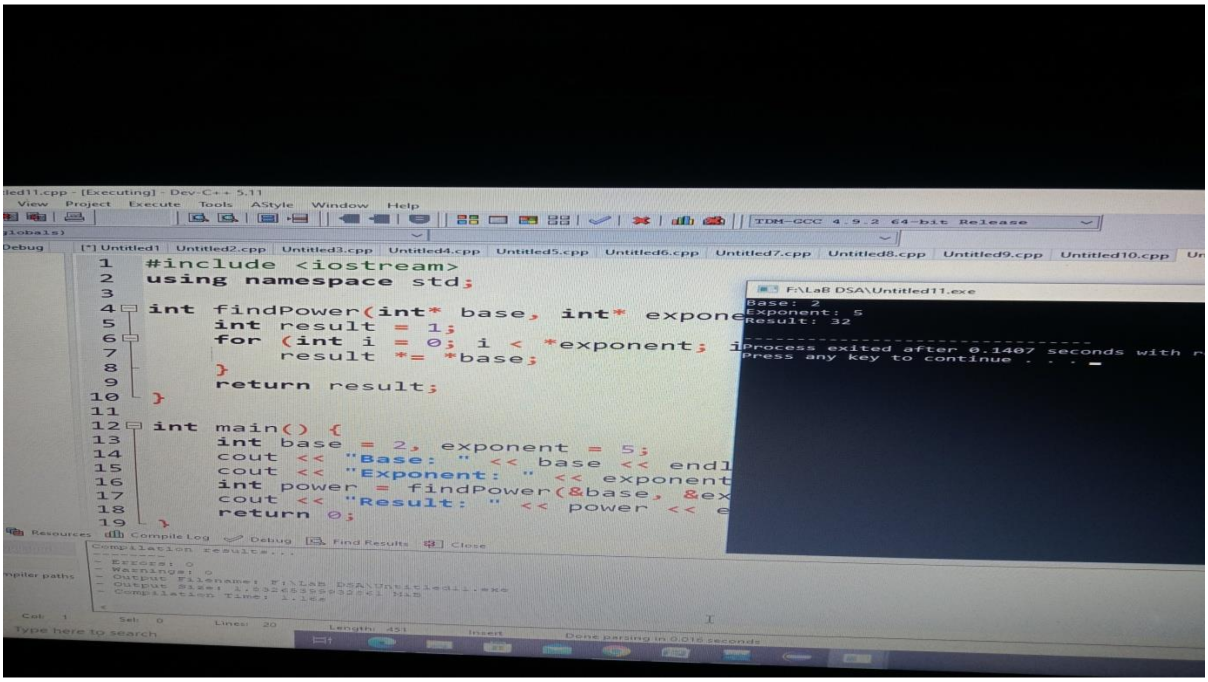
```
#include <iostream>
using namespace std;

double findAverage(int* arr, int size) {
    int sum = 0;
    for (int i = 0; i < size; i++) {
        sum += *(arr + i);
    }
    return static_cast<double>(sum) / size
}

int main() {
    int arr[] = {1, 2, 3, 4, 5};
    int size = sizeof(arr) / sizeof(arr[0]
    double average = findAverage(arr, size
    cout << "Average of elements in the ar
    return 0;
}
```

```
Average of elements in the array: 3

--------------------------------
Process exited after 0.1479 seconds with
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Untitled10.exe
- Output Size: 1.83265781402588 MiB
- Compilation Time: 0.91s
```

11.Write a C++ program to find the power of a number by using pointers

```
#include <iostream>
using namespace std;

int findPower(int* base, int* exponent) {
    int result = 1;
    for (int i = 0; i < *exponent; i++) {
        result *= *base;
    }
    return result;
}

int main() {
    int base = 2, exponent = 5;
    cout << "Base: " << base << endl;
    cout << "Exponent: " << exponent << endl;
    int power = findPower(&base, &exponent);
    cout << "Result: " << power << endl;
    return 0;
}
```



led11.cpp - [Executing] - Dev-C++ 5.11
View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
globals)
Debug
[*] Untitled1 Untitled2.cpp Untitled3.cpp Untitled4.cpp Untitled5.cpp Untitled6.cpp Untitled7.cpp Untitled8.cpp Untitled9.cpp Untitled10.cpp
Resources Compile Log Debug Find Results Close
Col: 1 Sel: 0 Lines: 20 Length: 451 Insert Done parsing in 0.016 seconds
Type here to search
F:\LaB DSA\Untitled11.exe

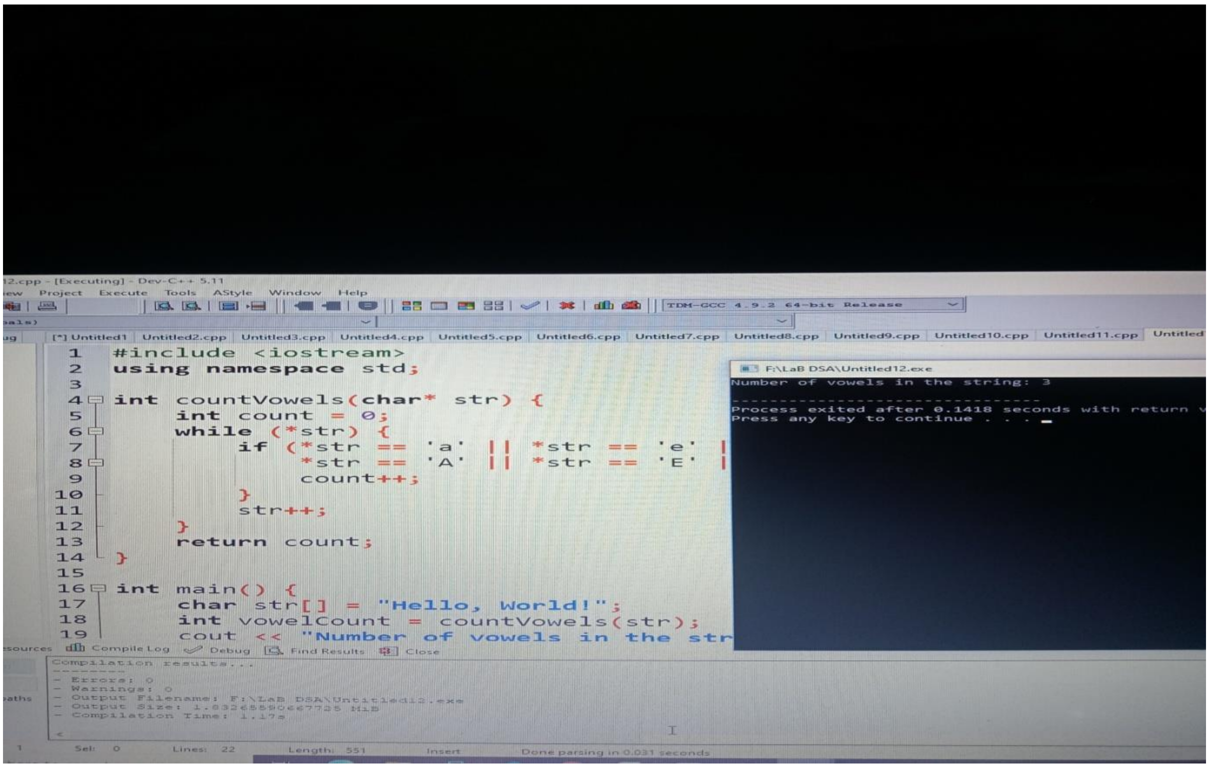
```
#include <iostream>
using namespace std;

int findPower(int* base, int* expone
        int result = 1;
        for (int i = 0; i < *exponent; i
            result *= *base;
        }
        return result;
}

int main() {
        int base = 2, exponent = 5;
        cout << "Base: " << base << endl
        cout << "Exponent: " << exponent
        int power = findPower(&base, &ex
        cout << "Result: " << power << e
        return 0;
}
```

```
Base: 2
Exponent: 5
Result: 32

--------------------------------
Process exited after 0.1407 seconds with r
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Untitled11.exe
- Output Size: 1.83265399932861 MiB
- Compilation Time: 1.16s
```

12.Write a C++ program to find the number of vowels in a string by using pointers

```
#include <iostream>
using namespace std;

int countVowels(char* str) {
    int count = 0;
    while (*str) {
        if (*str == 'a' || *str == 'e' || *str == 'i' || *str == 'o' || *str == 'u' ||
            *str == 'A' || *str == 'E' || *str == 'I' || *str == 'O' || *str == 'U') {
            count++;
        }
        str++;
    }
    return count;
}

int main() {
    char str[] = "Hello, World!";
    int vowelCount = countVowels(str);
    cout << "Number of vowels in the string: " << vowelCount << endl;
    return 0;
}
```



```
#include <iostream>
using namespace std;

int countVowels(char* str) {
    int count = 0;
    while (*str) {
        if (*str == 'a' || *str == 'e' |
            *str == 'A' || *str == 'E' |
            count++;
        }
        str++;
    }
    return count;
}

int main() {
    char str[] = "Hello, World!";
    int vowelCount = countVowels(str);
    cout << "Number of vowels in the str
```

F:\LaB DSA\Untitled12.exe

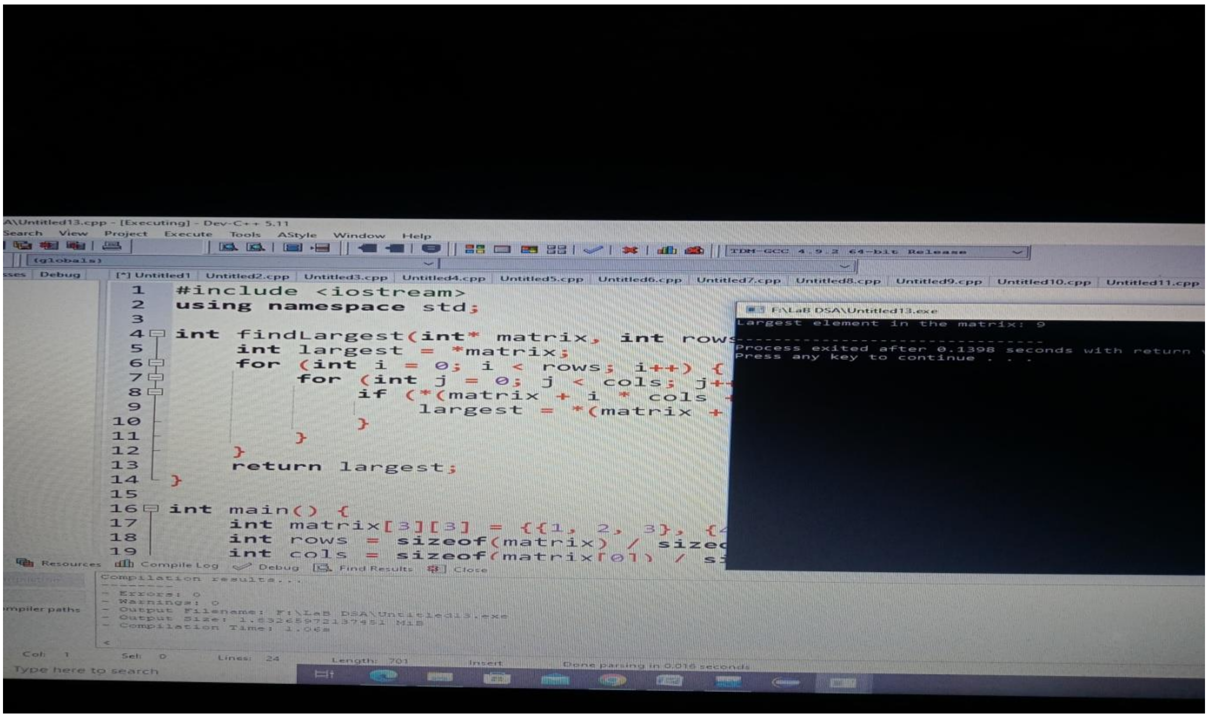
```
Number of vowels in the string: 3

--------------------------------
Process exited after 0.1418 seconds with return v
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Untitled12.exe
- Output Size: 1.83265590667725 MiB
- Compilation Time: 1.17s
```

1 Sel: 0 Lines: 22 Length: 551 Insert Done parsing in 0.031 seconds

13.Write a C++ program to find the largest element in an array by using pointers

```
#include <iostream>
using namespace std;

int findLargest(int* matrix, int rows, int cols) {
    int largest = *matrix;
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
            if (*(matrix + i * cols + j) > largest) {
                largest = *(matrix + i * cols + j);
            }
        }
    }
    return largest;
}

int main() {
    int matrix[3][3] = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};
    int rows = sizeof(matrix) / sizeof(matrix[0]);
    int cols = sizeof(matrix[0]) / sizeof(matrix[0][0]);
    int largest = findLargest(&matrix[0][0], rows, cols);
    cout << "Largest element in the matrix: " << largest << endl;
    return 0;
}
```



A\Untitled13.cpp - [Executing] - Dev-C++ 5.11
Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
[*] Untitled1 Untitled2.cpp Untitled3.cpp Untitled4.cpp Untitled5.cpp Untitled6.cpp Untitled7.cpp Untitled8.cpp Untitled9.cpp Untitled10.cpp Untitled11.cpp
F:\LaB DSA\Untitled13.exe
Resources Compile Log Debug Find Results Close
Col: 1 Sel: 0 Lines: 24 Length: 701 Insert Done parsing in 0.016 seconds
Type here to search

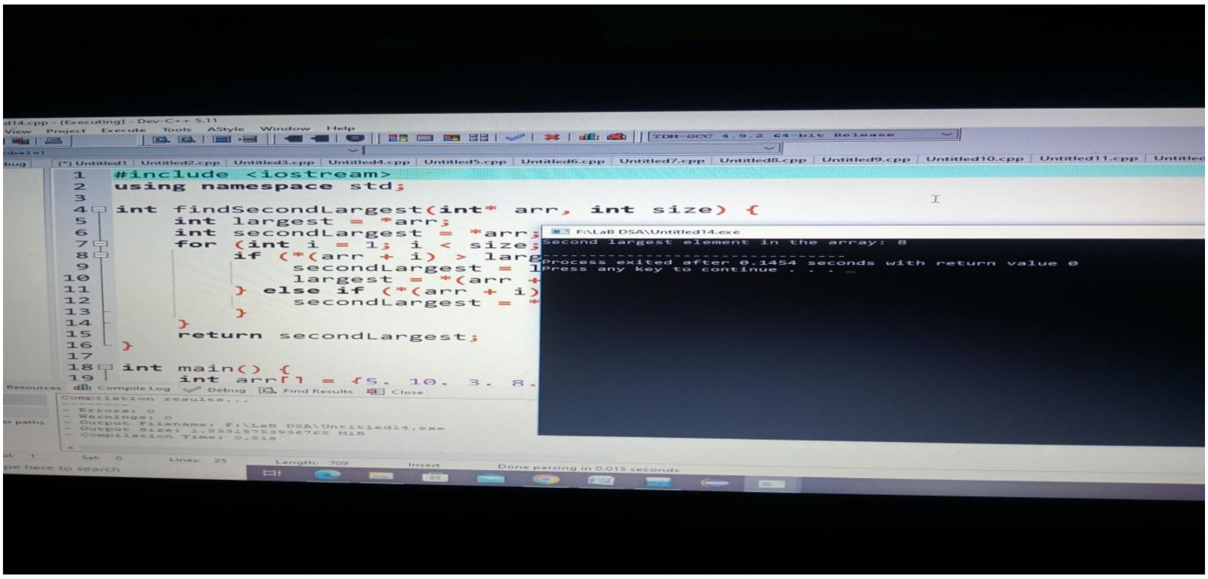
```
#include <iostream>
using namespace std;

int findLargest(int* matrix, int rows
    int largest = *matrix;
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++
            if (*(matrix + i * cols +
                largest = *(matrix +
            }
        }
    }
    return largest;
}

int main() {
    int matrix[3][3] = {{1, 2, 3}, {4
    int rows = sizeof(matrix) / sizeo
    int cols = sizeof(matrix[0]) / si
```

```
Largest element in the matrix: 9
--------------------------------
Process exited after 0.1398 seconds with return
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Untitled13.exe
- Output Size: 1.83265972137451 MiB
- Compilation Time: 1.06s
```

14.Write a C++ program to find the second largest element in an array by using pointers

```cpp
#include <iostream>
using namespace std;

int findSecondLargest(int* arr, int size) {
    int largest = *arr;
    int secondLargest = *arr;
    for (int i = 1; i < size; i++) {
        if (*(arr + i) > largest) {
            secondLargest = largest;
            largest = *(arr + i);
        } else if (*(arr + i) > secondLargest && *(arr + i) != largest) {
            secondLargest = *(arr + i);
        }
    }
    return secondLargest;
}

int main() {
    int arr[] = {5, 10, 3, 8, 2};
    int size = sizeof(arr) / sizeof(arr[0]);
    int secondLargest = findSecondLargest(arr, size);
    cout << "Second largest element in the array: " << secondLargest << endl;
    return 0;
}
```



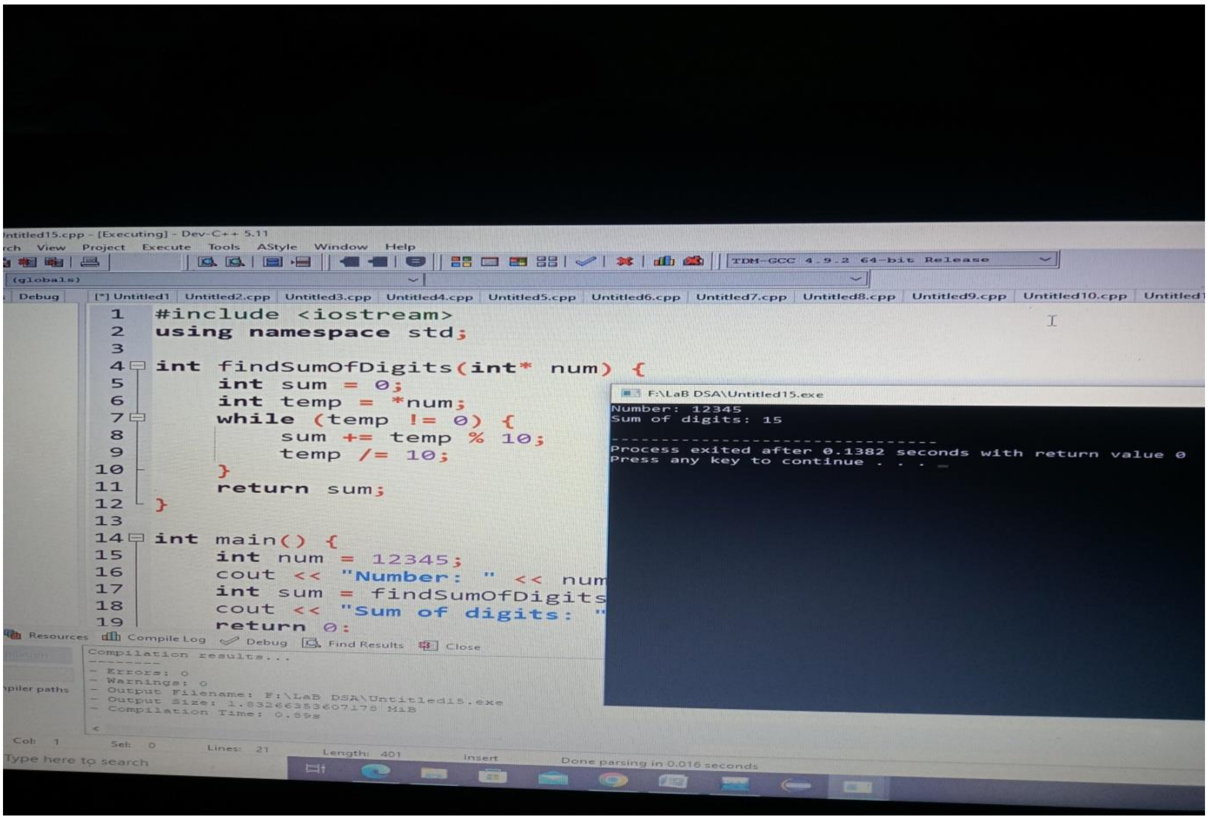
F:\LaB DSA\Untitled14.exe

```
Second largest element in the array: 8
--------------------------------
Process exited after 0.1454 seconds with return value 0
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Untitled14.exe
- Output Size: 1.83315753936768 MiB
- Compilation Time: 0.21s
```

15. Write a C++ program to find the sum of a digit of a number by using pointers

```
#include <iostream>
using namespace std;

int findSumOfDigits(int* num) {
    int sum = 0;
    int temp = *num;
    while (temp != 0) {
        sum += temp % 10;
        temp /= 10;
    }
    return sum;
}

int main() {
    int num = 12345;
    cout << "Number: " << num << endl;
    int sum = findSumOfDigits(&num);
    cout << "Sum of digits: " << sum << endl;
    return 0;
}
```



ntitled15.cpp - [Executing] - Dev-C++ 5.11
rch View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Debug
[*] Untitled1 Untitled2.cpp Untitled3.cpp Untitled4.cpp Untitled5.cpp Untitled6.cpp Untitled7.cpp Untitled8.cpp Untitled9.cpp Untitled10.cpp Untitled1
F:\LaB DSA\Untitled15.exe
Resources Compile Log Debug Find Results Close
piler paths
Col: 1 Sel: 0 Lines: 21 Length: 401 Insert Done parsing in 0.016 seconds
Type here to search

```
#include <iostream>
using namespace std;

int findSumOfDigits(int* num) {
    int sum = 0;
    int temp = *num;
    while (temp != 0) {
        sum += temp % 10;
        temp /= 10;
    }
    return sum;
}

int main() {
    int num = 12345;
    cout << "Number: " << num
    int sum = findSumOfDigits
    cout << "Sum of digits: "
    return 0:
```

```
Number: 12345
Sum of digits: 15

--------------------------------
Process exited after 0.1382 seconds with return value 0
Press any key to continue . . .
```

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: F:\LaB DSA\Untitled15.exe
- Output Size: 1.83266353607178 MiB
- Compilation Time: 0.89s
```