Due before:	**11:59 PM (before Midnight) on Wednesday April 29, 2020**

Introduction to Systems Programming (System I)

Lab #8

Max Points: 50 (40 + 10)

Due: Wednesday April 29, 2020 before 11:59 PM

Email-based help Cutoff: 5:00 PM on Tue, April 28, 2020

Objective: The objective of this exercise is to gain experience with:

- Pointers and Memory
- Continue using CODE plugin for submission

Fill in answers to all of the questions. For some of the questions you can simply copy-paste appropriate text from the terminal/output window into this document. You may discuss the questions with your instructor.

Use NetBeans for this Lab

Name: **Ben Hilger**

Part #1: Pointers and Pointer Arithmetic

[15 points]

1. What type of data does a pointer store? **[1 point]**

   Pointers store memory address that point another variable of the same data type.

2. What information tells us the amount by which a pointer will increment? **[1 point]**

   The datatype of the pointer tells us the amount by which a pointer will increment

3. What will `cout << ptr << *ptr << endl;` print? What is the difference between `ptr` and `*ptr` in that line of code? **[4 point]**

   It will print first the memory address and then the value that the pointer points to.

   The difference between ptr and *ptr is that ptr references the memory address that it's pointing to, and. *ptr references the value of the memory address.

4. What is the output from the following statement: **[2 point]**

```
int primes[] = {2,3,5,7,11};
int *ptr = primes;
ptr += 2;
cout << *ptr << endl;
```

The output written to `cout` is: 5

5. What is the output from the following statement: **[2 point]**

```
int odds[] = {1,3,5,7,11,13};
int *ptr = &odds[2];
cout << *(--ptr) << endl;
```

The output written to `cout` is: 3

6. What is the output from the following statement: **[2 point]**

```
int odds[] = {1,3,5,7,11,13};
int *ptr = &odds[2];
if( *ptr == 5){
     cout << *(ptr--) << endl;
} else{
     cout << *(++ptr) << endl;
}
```

The output written to `cout` is: 5

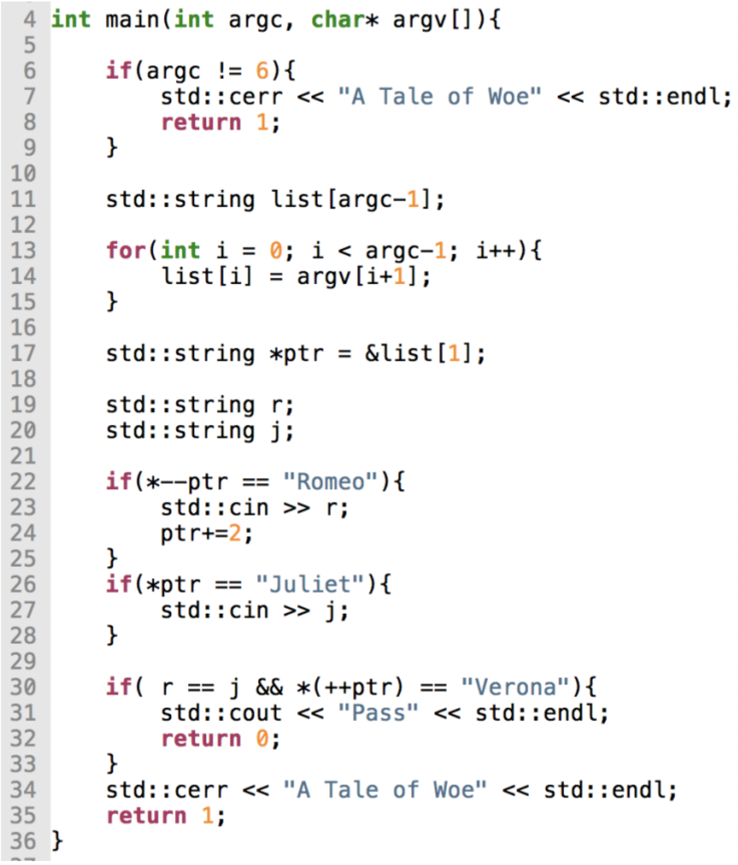
7. What is the output from the following statement: **[3 point]**

```
int odds[] = {1,3,5,7,11,13};
int primes[] = {2,3,5,7,11};
int *ptr1 = &odds[3];
int *ptr2 = &primes[3];
if( ptr1 == ptr2){
    cout << *(--ptr1) << endl;
} else{
    cout << *ptr1 << endl;
}
```

The output written to `cout` is: 7

Part #2: Code Analysis

[18 points]

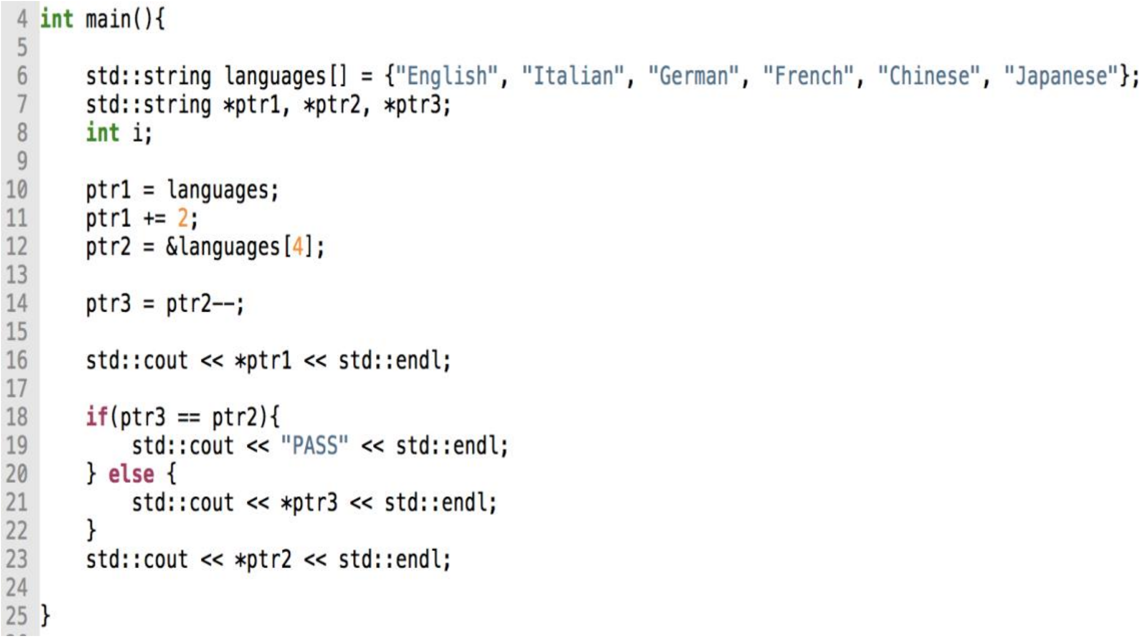


```
int main(int argc, char* argv[]){

    if(argc != 6){
        std::cerr << "A Tale of Woe" << std::endl;
        return 1;
    }

    std::string list[argc-1];

    for(int i = 0; i < argc-1; i++){
        list[i] = argv[i+1];
    }

    std::string *ptr = &list[1];

    std::string r;
    std::string j;

    if(*--ptr == "Romeo"){
        std::cin >> r;
        ptr+=2;
    }
    if(*ptr == "Juliet"){
        std::cin >> j;
    }

    if( r == j && *(++ptr) == "Verona"){
        std::cout << "Pass" << std::endl;
        return 0;
    }
    std::cerr << "A Tale of Woe" << std::endl;
    return 1;
}
```

1. What input would make lead the program (myProgram1) to output “Pass”? **[5 points]**

Argc = 6
Argv = [“”, “”, “Verona”, “‘, “”]

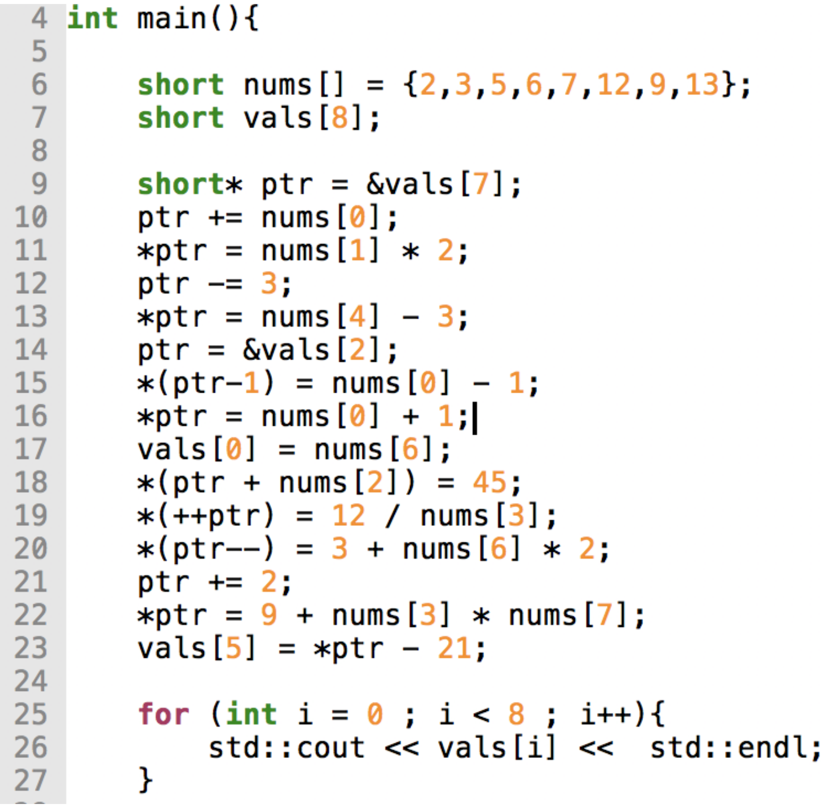
This keeps the strings r and j null (and therefore equal) and since the ++ptr is also pointing at Veronika, the if statement will hold true and the programing will output “Pass”



```
int main(){

    std::string languages[] = {"English", "Italian", "German", "French", "Chinese", "Japanese"};
    std::string *ptr1, *ptr2, *ptr3;
    int i;

    ptr1 = languages;
    ptr1 += 2;
    ptr2 = &languages[4];

    ptr3 = ptr2--;

    std::cout << *ptr1 << std::endl;

    if(ptr3 == ptr2){
        std::cout << "PASS" << std::endl;
    } else {
        std::cout << *ptr3 << std::endl;
    }
    std::cout << *ptr2 << std::endl;

}
```

2. What will the program print? **[5 points]**

German
Chinese
French



```
 4 int main(){
 5
 6     short nums[] = {2,3,5,6,7,12,9,13};
 7     short vals[8];
 8
 9     short* ptr = &vals[7];
10     ptr += nums[0];
11     *ptr = nums[1] * 2;
12     ptr -= 3;
13     *ptr = nums[4] - 3;
14     ptr = &vals[2];
15     *(ptr-1) = nums[0] - 1;
16     *ptr = nums[0] + 1;
17     vals[0] = nums[6];
18     *(ptr + nums[2]) = 45;
19     *(++ptr) = 12 / nums[3];
20     *(ptr--) = 3 + nums[6] * 2;
21     ptr += 2;
22     *ptr = 9 + nums[3] * nums[7];
23     vals[5] = *ptr - 21;
24
25     for (int i = 0 ; i < 8 ; i++){
26         std::cout << vals[i] <<  std::endl;
27     }
```

3. What will the program print? **[8 points]**

This program will throw a stack overflow exception at line 11 because the pointer points outside the 8 values of the short array.

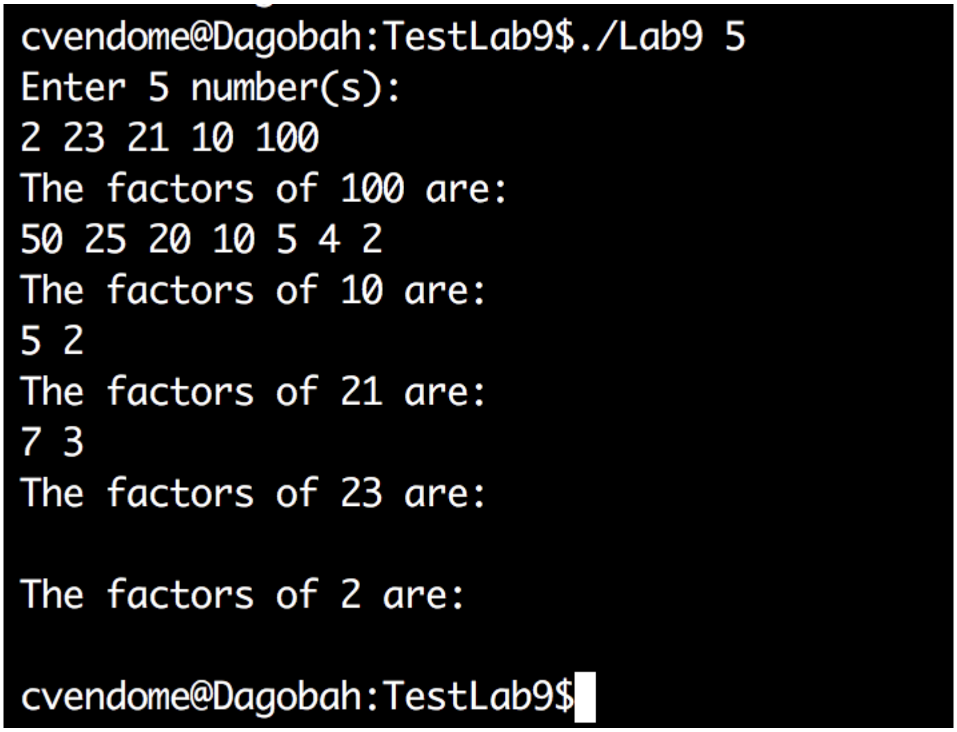
Part #3: Using Pointers to Iterate Loops
[10 points]

Program: In this lab, the goal is to write a program that utilizes points to traverse an array of numbers specified by a user. The program will get a set of numbers from standard input. It should use pointers to insert the numbers into the array and then use pointers to traverse the array in reverse while printing the memory address and the factors of the number located at that memory location to which it points. A sample output is provided. **The memory address and the even/odd should be comma separated and printed to standard output.**

The program will:

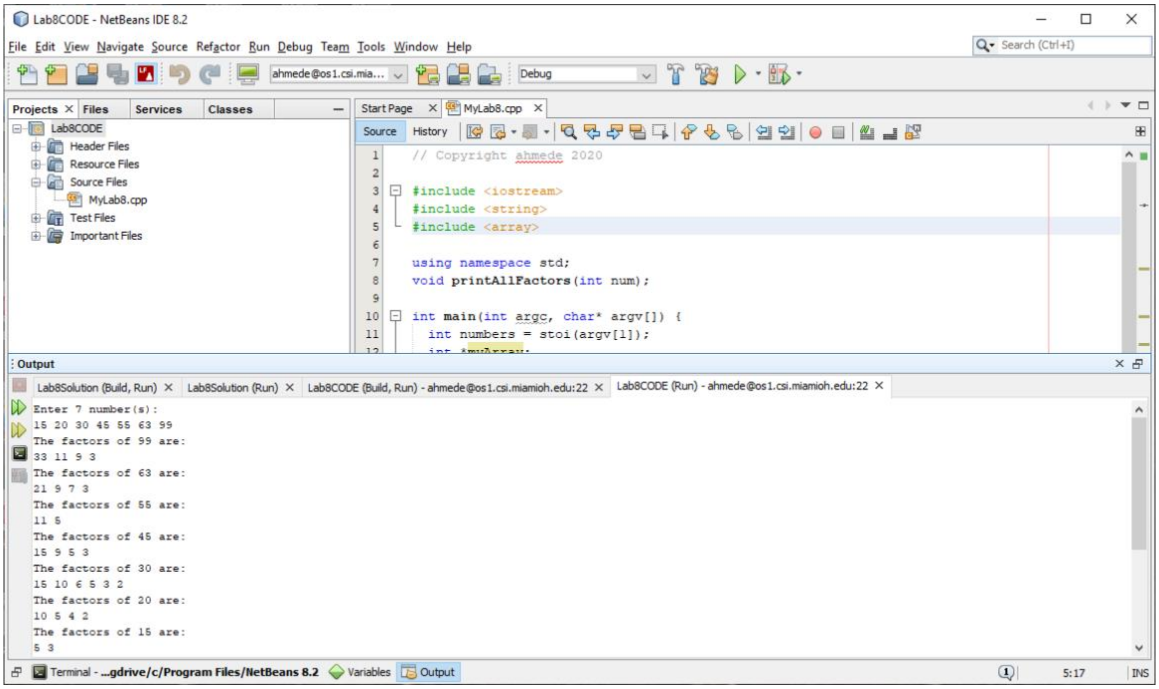
1. Read one command-line argument: the number of elements in the array **[1 point]**
   a. Example, the following command would ask the user for 5 numbers:
      ./lab8 5
2. Insert the numbers entered by the user into an integer array using pointers. **[2 points]**
3. Traverse the list in reverse using pointers. **[2 points]**
4. Print the value at the memory address to which it points and then print the factors of the value at the memory address to which it points (sample output below). **[2 points]**
   a. **NOTE: The output should *not* contain 1 and the number itself**
5. ***Code submission MUST be done through the code plug-in on Canvas for credit (instructions on the next page)***
6. Pass all the Test Cases [3 points]

SAMPLE OUTPUT:



```
cvendome@Dagobah:TestLab9$./Lab9 5
Enter 5 number(s):
2 23 21 10 100
The factors of 100 are:
50 25 20 10 5 4 2
The factors of 10 are:
5 2
The factors of 21 are:
7 3
The factors of 23 are:

The factors of 2 are:

cvendome@Dagobah:TestLab9$
```

NOTE: Operating on the array without using pointers will NOT receive partial credit.



Lab8CODE - NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Team Tools Window Help
Search (Ctrl+I)
ahmede@os1.csi.mia...
Debug
Projects Files Services Classes
Lab8CODE
Header Files
Resource Files
Source Files
MyLab8.cpp
Test Files
Important Files
Start Page
MyLab8.cpp
Source History
Output
Lab8Solution (Build, Run)
Lab8Solution (Run)
Lab8CODE (Build, Run) - ahmede@os1.csi.miamioh.edu:22
Lab8CODE (Run) - ahmede@os1.csi.miamioh.edu:22
Terminal - ...gdrive/c/Program Files/NetBeans 8.2
Variables
Output
5:17
INS

```
// Copyright ahmede 2020

#include <iostream>
#include <string>
#include <array>

using namespace std;
void printAllFactors(int num);

int main(int argc, char* argv[]) {
  int numbers = stoi(argv[1]);
  int *myArray;
```

```
Enter 7 number(s):
15 20 30 45 55 63 99
The factors of 99 are:
33 11 9 3
The factors of 63 are:
21 9 7 3
The factors of 55 are:
11 5
The factors of 45 are:
15 9 5 3
The factors of 30 are:
15 10 6 5 3 2
The factors of 20 are:
10 5 4 2
The factors of 15 are:
5 3
```

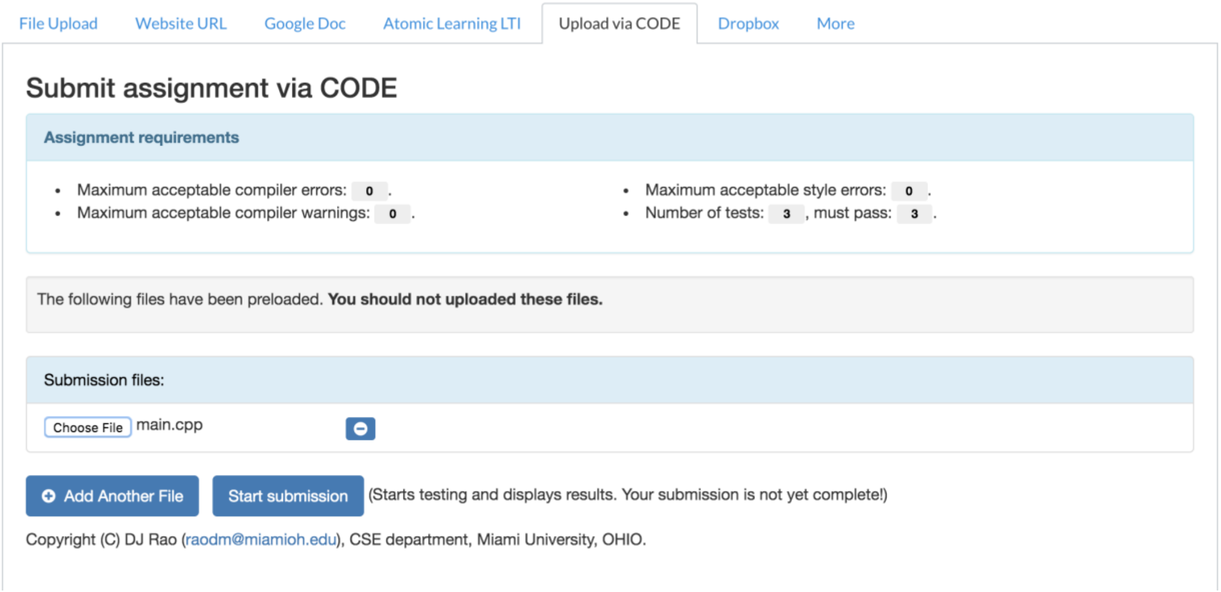
Submit to Canvas

There are **TWO** assignments on canvas – one for the program and one for code submission

1. The document must be submitted as a PDF
2. The code must be submitted through the Code Plug-in

Submit to Code Plug-in

To submit, click the “Upload via CODE” tab then click “Choose File” to upload the source code



Submit assignment via CODE

Assignment requirements

- Maximum acceptable compiler errors: 0 .
- Maximum acceptable compiler warnings: 0 .
- Maximum acceptable style errors: 0 .
- Number of tests: 3 , must pass: 3 .

The following files have been preloaded. **You should not uploaded these files.**

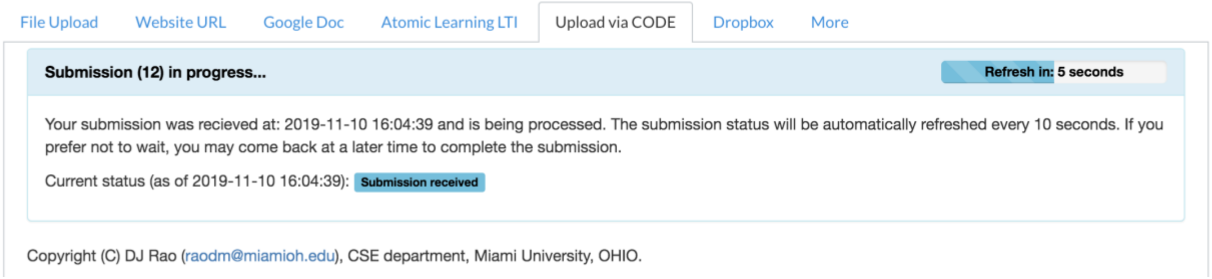
Submission files:

Choose File main.cpp

(Starts testing and displays results. Your submission is not yet complete!)

Copyright (C) DJ Rao (raodm@miamioh.edu), CSE department, Miami University, OHIO.

It will then submit the file



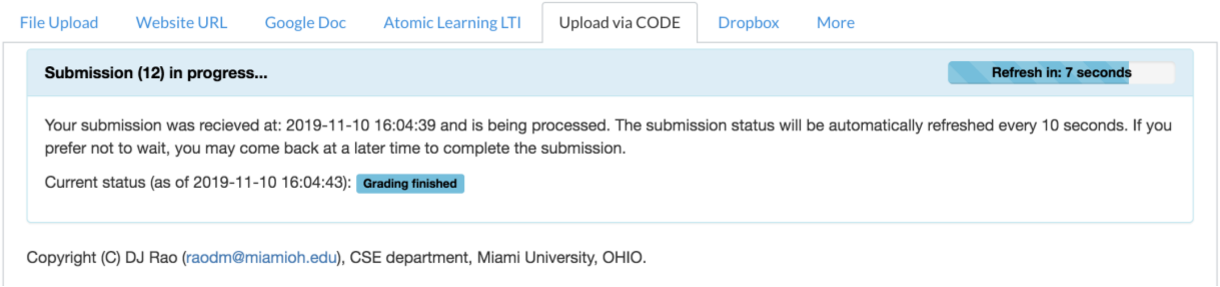
Submission (12) in progress...

Your submission was recieved at: 2019-11-10 16:04:39 and is being processed. The submission status will be automatically refreshed every 10 seconds. If you prefer not to wait, you may come back at a later time to complete the submission.

Current status (as of 2019-11-10 16:04:39): **Submission received**

Copyright (C) DJ Rao (raodm@miamioh.edu), CSE department, Miami University, OHIO.

And subsequently run test cases on the program



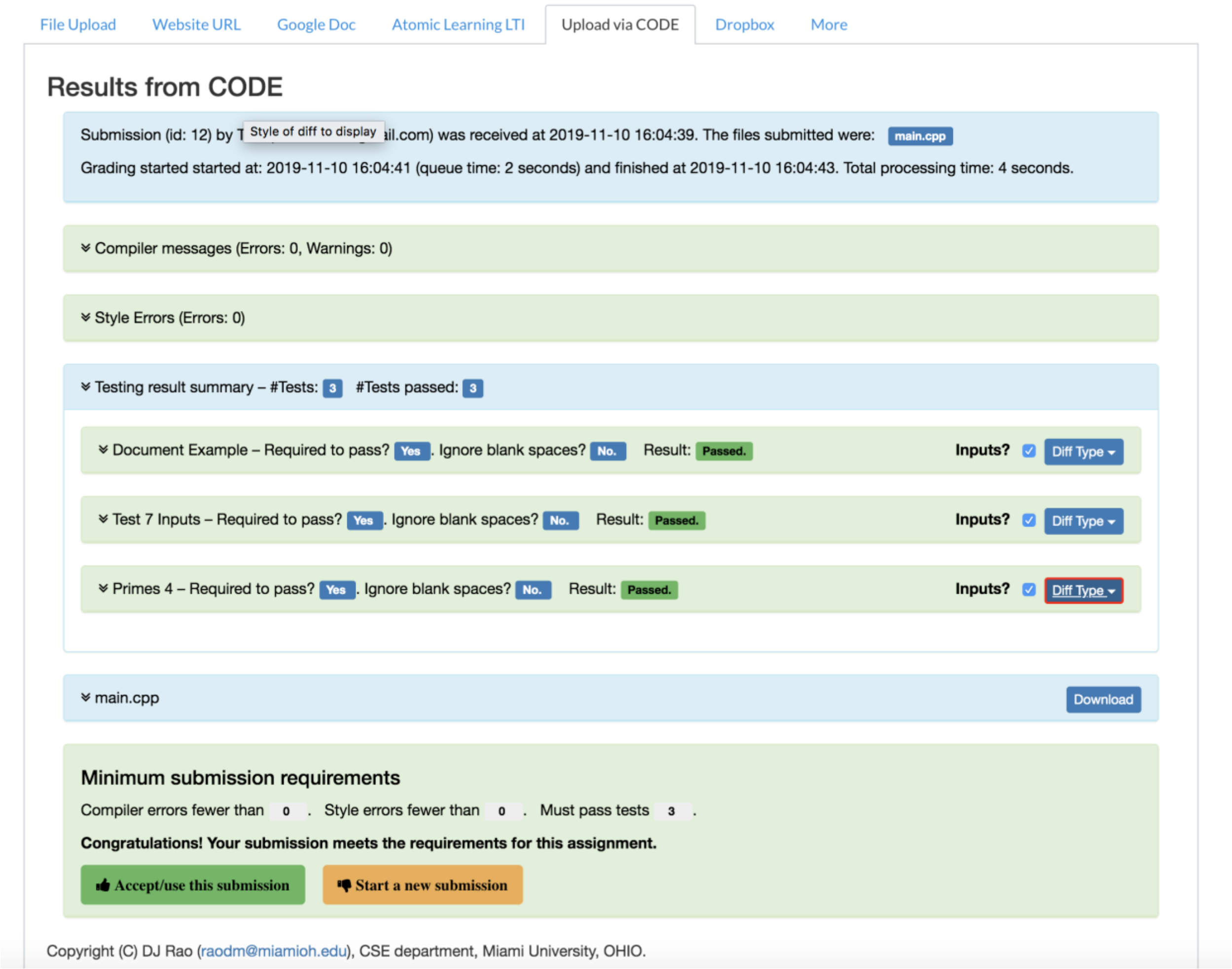
Submission (12) in progress...

Your submission was recieved at: 2019-11-10 16:04:39 and is being processed. The submission status will be automatically refreshed every 10 seconds. If you prefer not to wait, you may come back at a later time to complete the submission.

Current status (as of 2019-11-10 16:04:43): **Grading finished**

Copyright (C) DJ Rao (raodm@miamioh.edu), CSE department, Miami University, OHIO.

The results of the tests will be displayed and can be expanded:



Results from CODE

Submission (id: 12) by T Style of diff to display il.com) was received at 2019-11-10 16:04:39. The files submitted were: main.cpp

Grading started started at: 2019-11-10 16:04:41 (queue time: 2 seconds) and finished at 2019-11-10 16:04:43. Total processing time: 4 seconds.

» Compiler messages (Errors: 0, Warnings: 0)

» Style Errors (Errors: 0)

» Testing result summary – #Tests: 3 #Tests passed: 3

» Document Example – Required to pass? Yes. Ignore blank spaces? No. Result: Passed. Inputs? ☑ Diff Type ▾

» Test 7 Inputs – Required to pass? Yes. Ignore blank spaces? No. Result: Passed. Inputs? ☑ Diff Type ▾

» Primes 4 – Required to pass? Yes. Ignore blank spaces? No. Result: Passed. Inputs? ☑ Diff Type ▾

» main.cpp Download

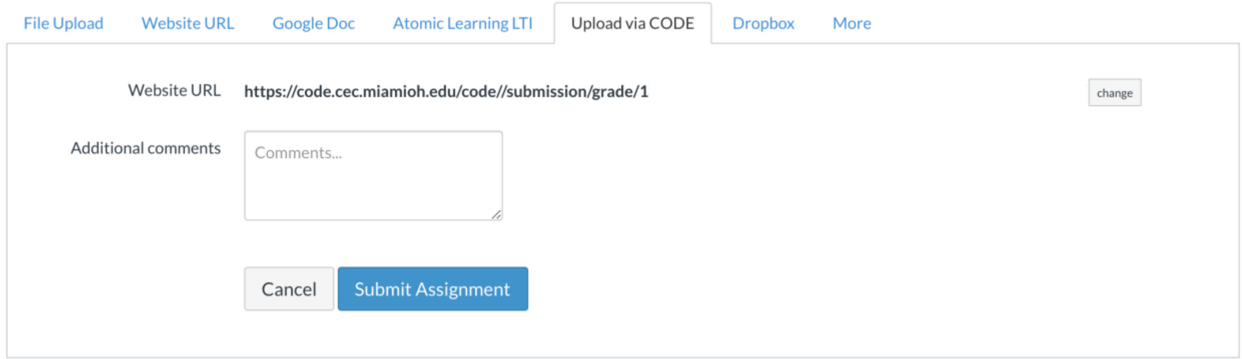
Minimum submission requirements

Compiler errors fewer than 0 . Style errors fewer than 0 . Must pass tests 3 .

Congratulations! Your submission meets the requirements for this assignment.

Copyright (C) DJ Rao (raodm@miamioh.edu), CSE department, Miami University, OHIO.

To submit, you must click "Accept/use this submission" and it will let you submit:



Website URL https://code.cec.miamioh.edu/code//submission/grade/1

Additional comments Comments...

Click submit assignment and it will submit the program.

- No late assignments will be accepted!
- This work is to be done individually
- The submission file will be saved with the name ***Lab8_yourMUID.pdf***
- Assignment is due before Midnight Wednesday April 29, 2020.
- On or before the due time, drop the *electronic copy* of your work in the *canvas*
- Don't forget to Turn in the files!   Lab8_yourMUID.pdf Lab8_yourMUID*.cpp.