**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. |

1. 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 |

1. 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. |

1. 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 | |

1. 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 |

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

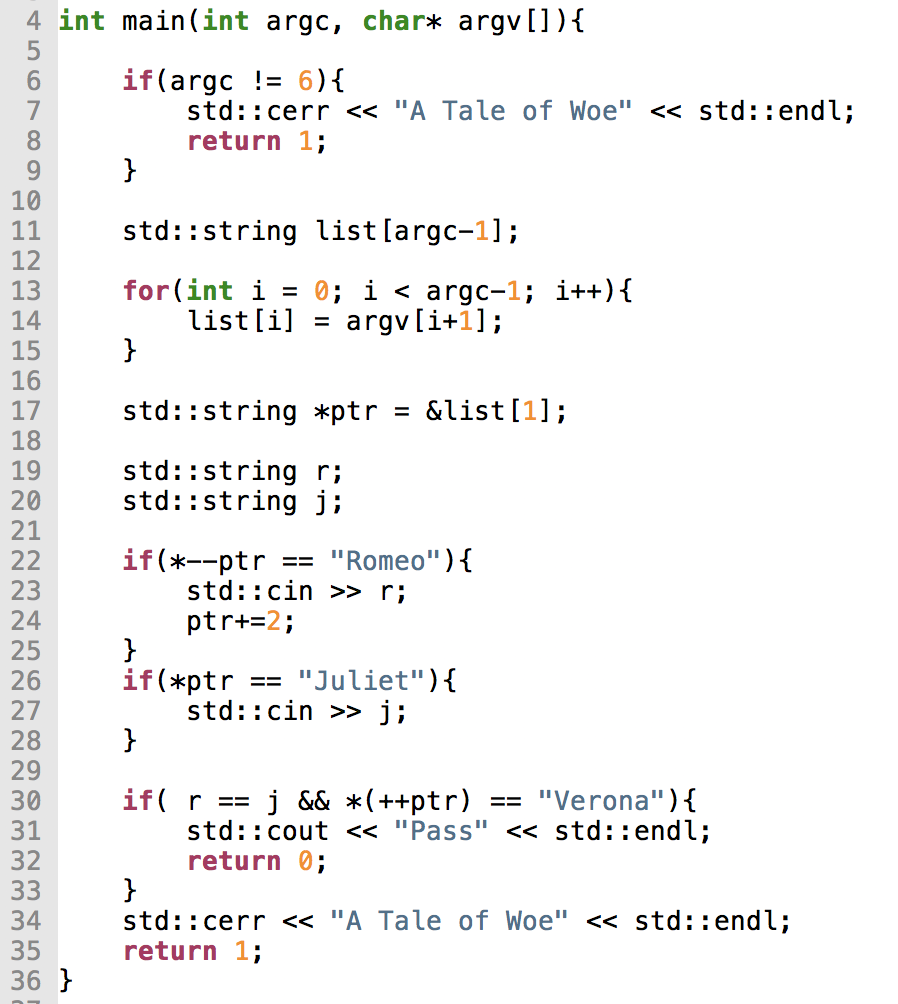
|  |  |
| --- | --- |
| int odds[] = {1,3,5,7,11,13};  int \*ptr = &odds[2];  if( \*ptr == 5){  cout << \*(ptr--) << endl;  } else{  cout << \*(++ptr) << endl;  } |  |
| The output written to cout is: | 5 |

1. 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 | |

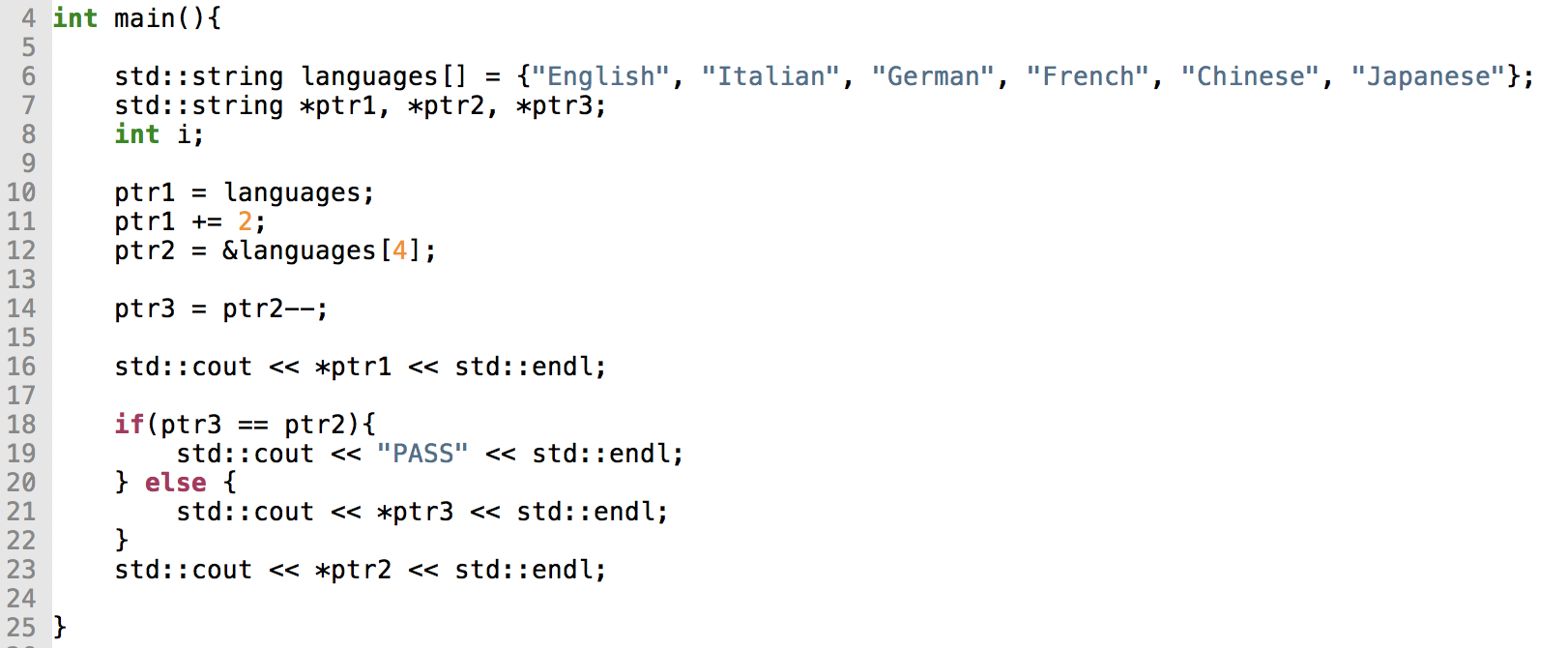
# Part #2: Code Analysis

**[18 points]**

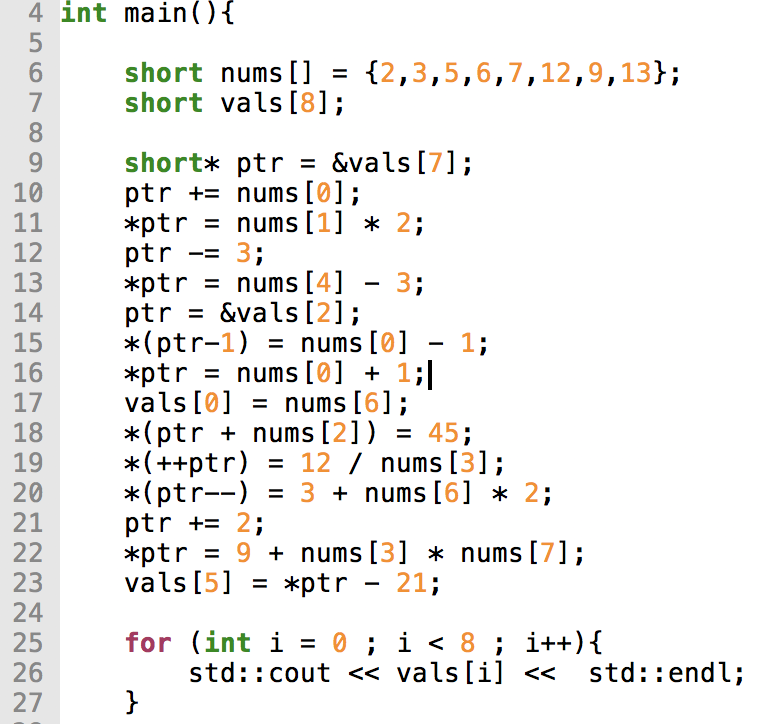
1. 

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” |

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

|  |
| --- |
| German  Chinese  French |

1. 

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.**

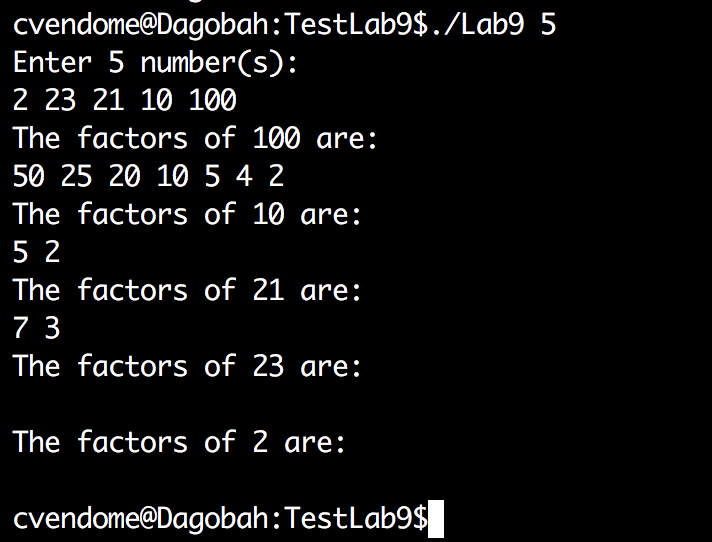
The program will:

1. Read one command-line argument: the number of elements in the array **[1 point]**
   1. Example, the following command would ask the user for 5 numbers:

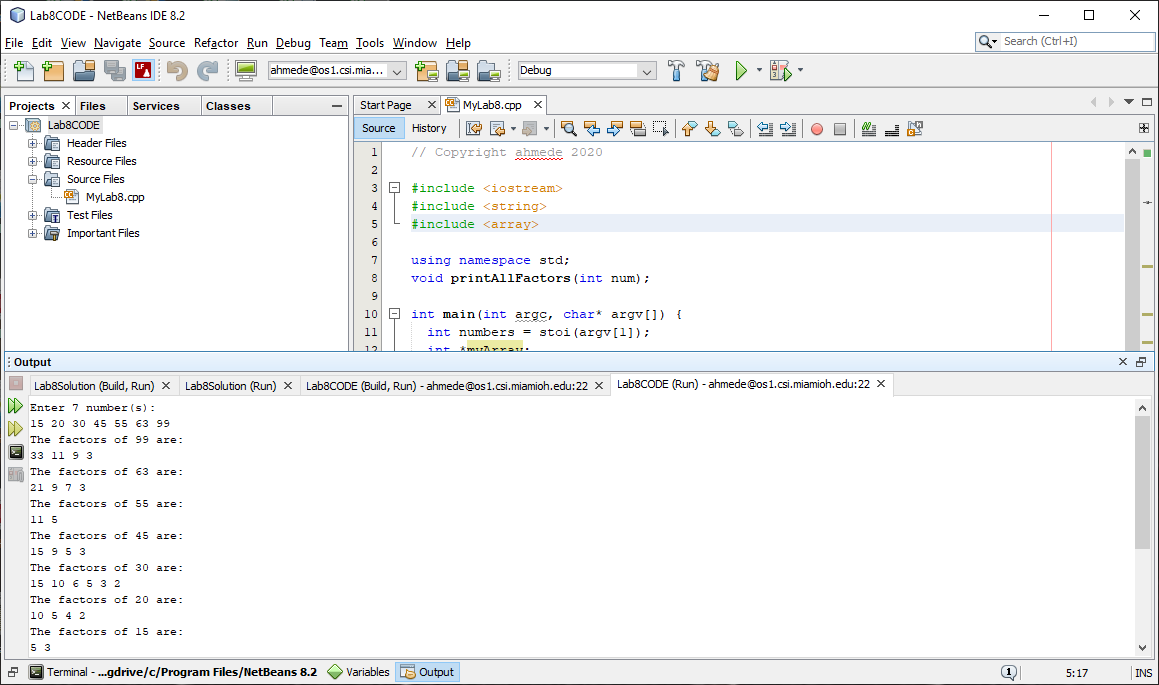
./lab8 5

1. Insert the numbers entered by the user into an integer array using pointers. **[2 points]**
2. Traverse the list in reverse using pointers. **[2 points]**
3. 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]**
   1. **NOTE: The output should *not* contain 1 and the number itself**
4. ***Code submission MUST be done through the code plug-in on Canvas for credit (instructions on the next page)***
5. Pass all the Test Cases [3 points]

SAMPLE OUTPUT:



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



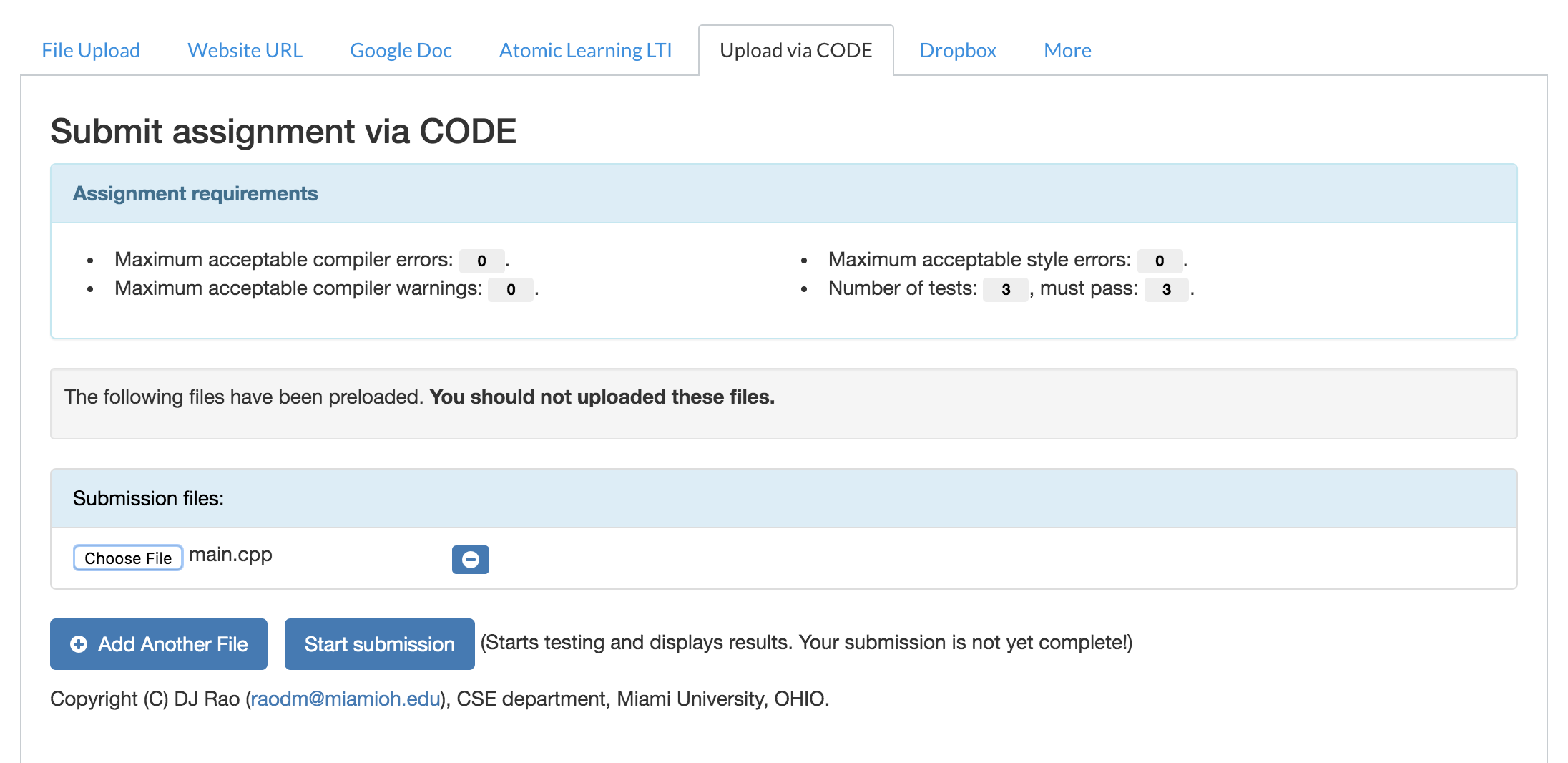
# Submit to Canvas

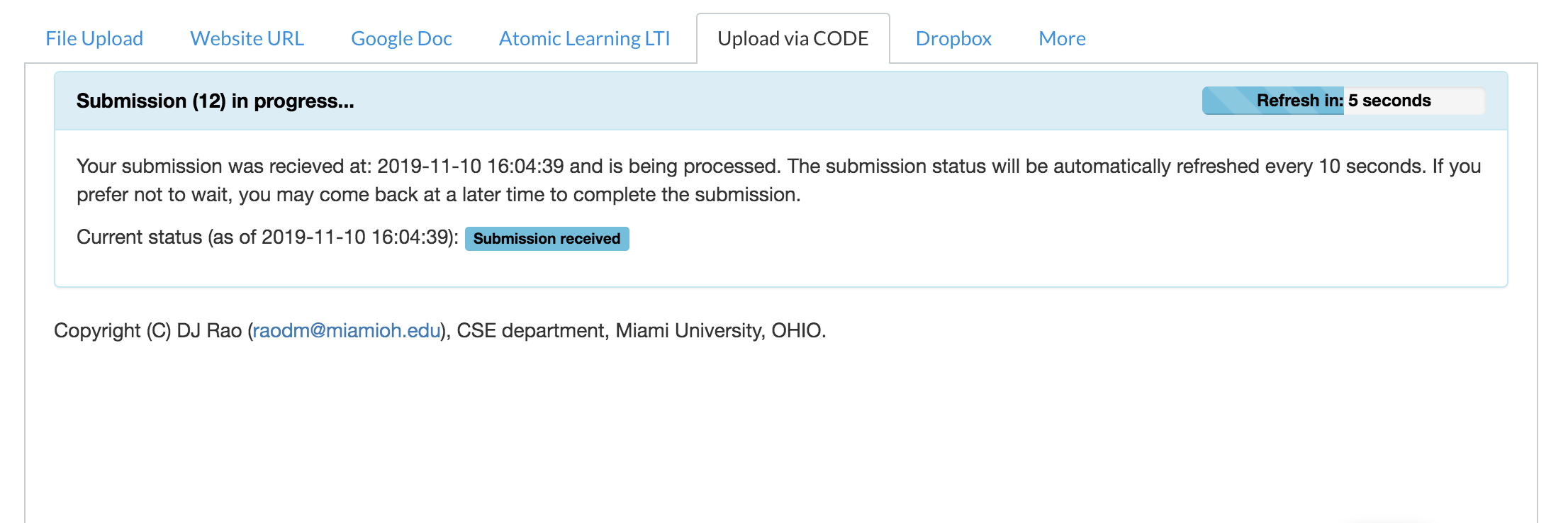
There are **TWO** assignments on canvas – one for the program and one for code submission

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

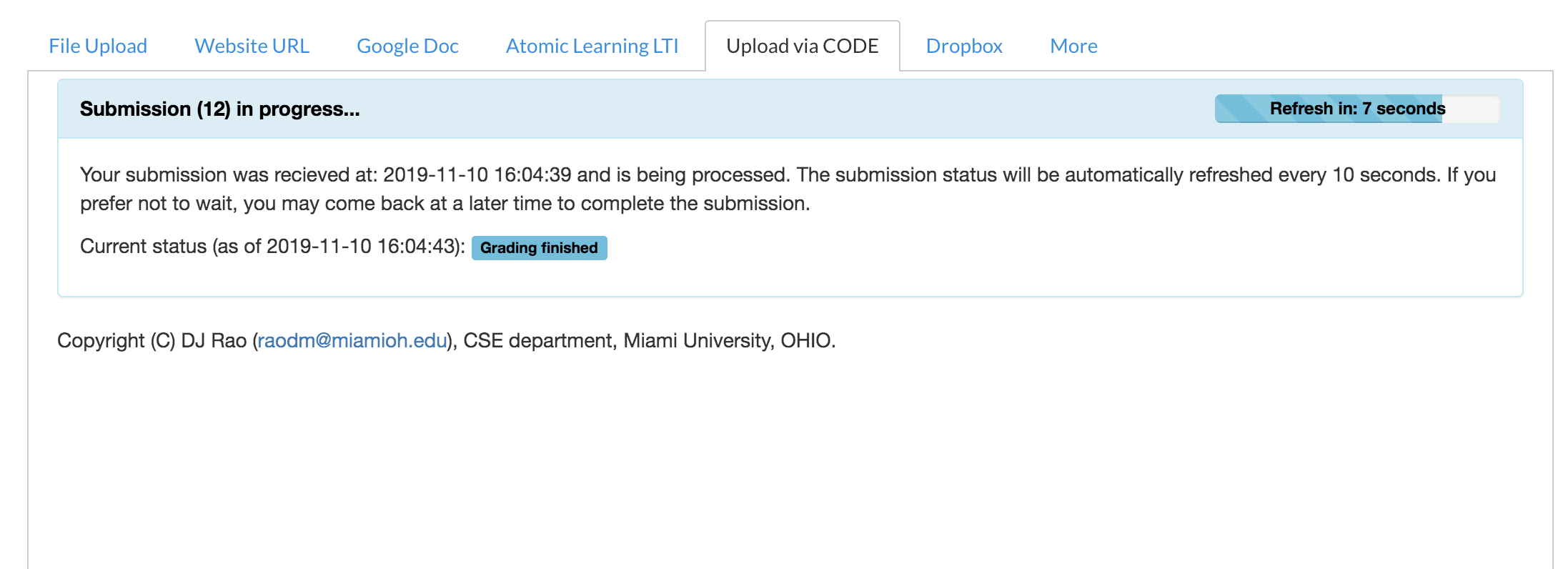
# Submit to Code Plug-in

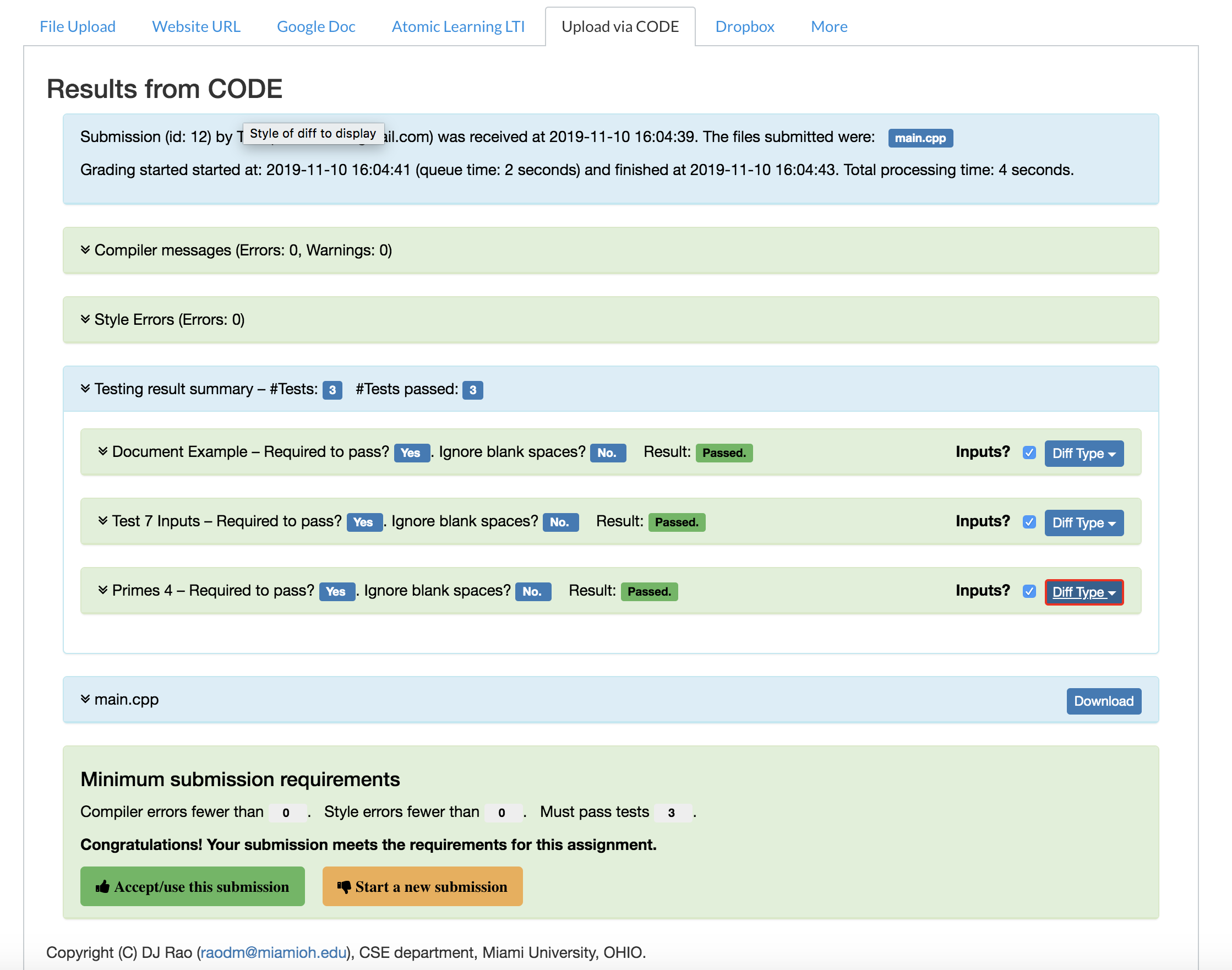
To submit, click the “Upload via CODE” tab then click “Choose File” to upload the source code



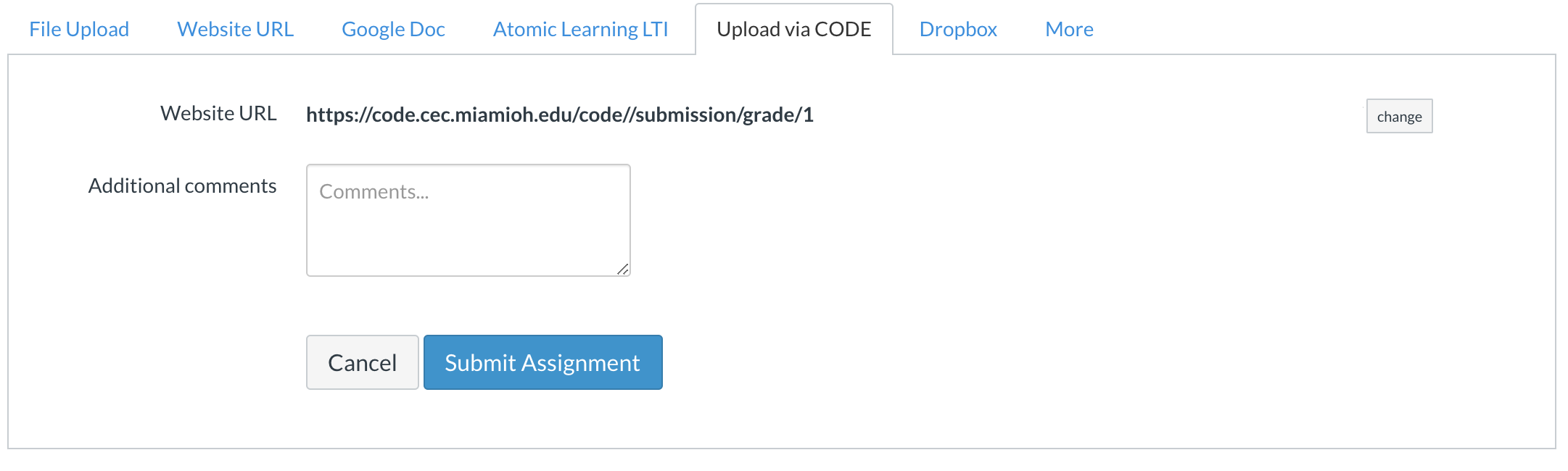
It will then submit the file

And subsequently run test cases on the program

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



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



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.