

**Welcome to CS 116! About the Course and Author**

Dr. Carmen Bruni obtained his Ph.D. in Mathematics from the University of British Columbia in 2015 with his thesis on *Twisted Extensions of Fermat's Last Theorem*, extending the seminal work of Andrew Wiles in the mid-90s. Since that time, he has worked at the University of Waterloo. He holds undergraduate degrees from the University of Waterloo, double majoring in Pure Mathematics and Computer Science and also obtained a master's degree in Pure Mathematics in 2010.

He has been teaching in the David R. Cheriton School of Computer Science since 2017 following a two year position with the Centre for Education in Mathematics and Computing. He has received numerous awards for teaching excellence including a prestigious Killam Graduate Teaching Award in 2013. His current research interests are in mathematics education and pedagogy. He may be reached at [cbruni@uwaterloo.ca](mailto:cbruni@uwaterloo.ca).

## Recommended Hardware and Software

### Recommended Hardware

Students are expected to use a laptop or desktop computer to access this platform. A Google Chromebook or tablet should also be sufficient provided you are using one of the web browsers listed below. Do not access this platform using a smart phone or other device as some functionality may not work properly.

### Recommended Web Browser

The edX platform works best with one of Google Chrome or Mozilla Firefox. Do not use other browsers like Safari, Microsoft Edge, or Internet Explorer as functionality of the site may be diminished.

**All of your work in CS 116 can be done within a browser and that is how we recommend you complete all the concept checks, quizzes and assignments.** There is no need for any additional software.

### Optional Offline IDE

If you strongly prefer to work offline for assignments, which we don't recommend for most students, then the preferred IDE is *Wing 101*. We cannot promise to provide support for debugging installation errors. This means that if you decide to download Wing 101, you should follow the instructions below carefully. There are two sets of instructions — one for Windows Users and one for Mac Users.

Additionally, if you are working offline, you must download the [check.py](#) module needed for testing on assignments. This is only necessary if you want to work offline. This file **must be in the same folder as the assignment problem you are working on**. This is the most common cause of the No module named check error.

### Instructions for Windows Users

If you are using a Mac operating system, see below.

- 1 Go to the Python Download page, which is located at <https://www.python.org/downloads>. You should see the following on the top of the page.



- 2 Click on the button that says **Download Python 3.7.2**. It should immediately start a download of Python. If you are redirected to another page, scroll all the way to the bottom and choose the download option that matches your Windows version.

- 3 Double click the file you have downloaded to install Python 3.7.2. It should open up an installation window like this one:.



You can follow the default settings for the installation. Make sure to allow Python to make changes to your computer for the installation to work.

Once the installation is done, make sure that you can find where your Python file is. If you can't, try going into the folder with your profile name and going into 'AppData/Local/Programs', and you should be able to find a folder named Python.

**Note:** If an error pops up at the end stating: "User cancelled installation". Then, when you rebegin your installation, uncheck the box for "Install launcher for all users (recommended)".

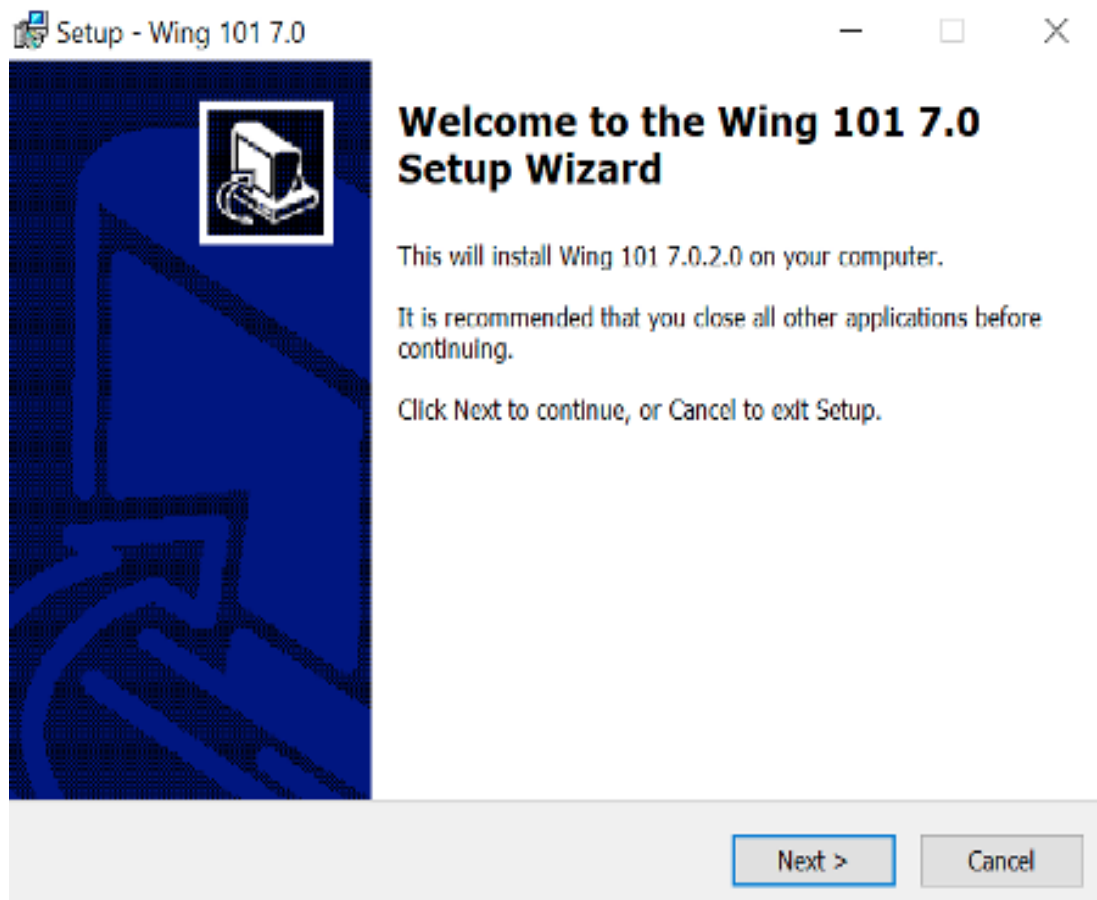
- 4 You will now be guided through the Wing 101 v7 installation process. Wing 101 is an Integrated Development Environment (IDE) like Dr. Racket. It allows you to write and run code in one program. Go to the Wing IDE v7 download page, which is located at <http://wingware.com/downloads/wingide-101>. Make sure that the top of the page says Wing 101 as this is the free version of Wing.

**Wing 101 - Version 7.0.1 - Released 2019-04-18**

- 5 On the right-hand side of the page, there will be two green buttons. Press on the top one (the one highlighted in yellow below) to download the installer.

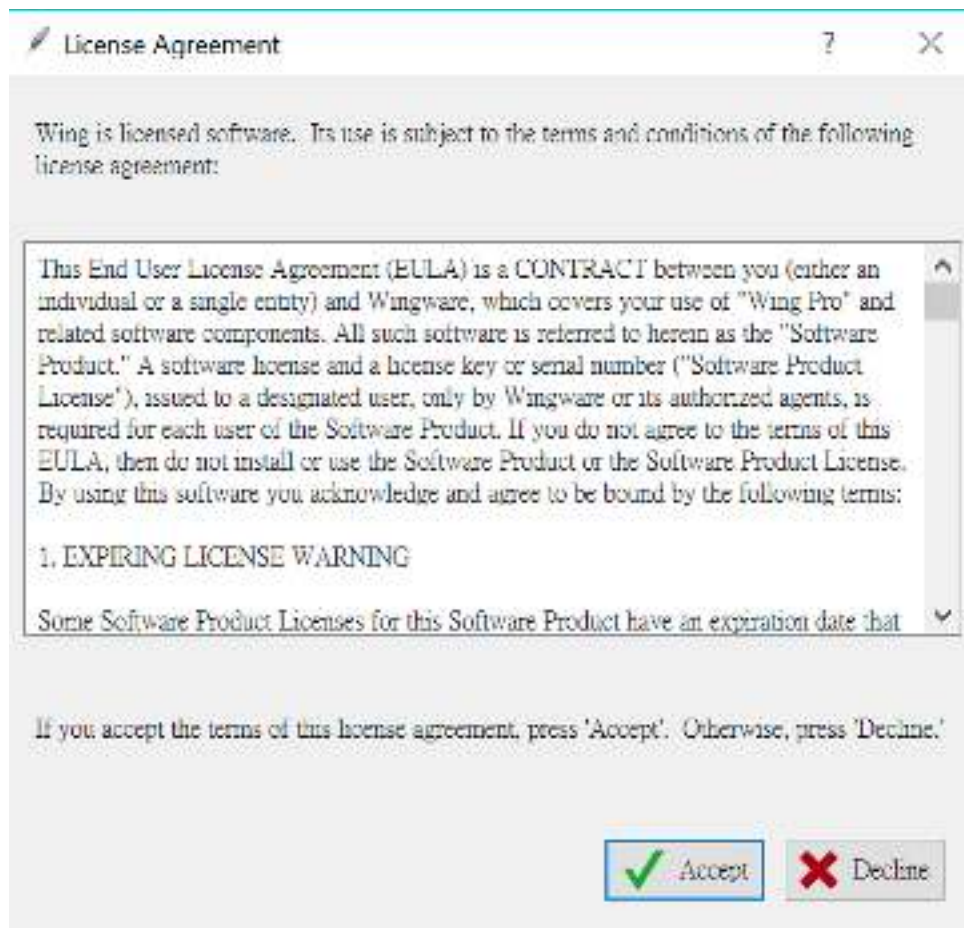


- 6 Once the installer is downloaded, double-click on it to open it. It should open a window like this.

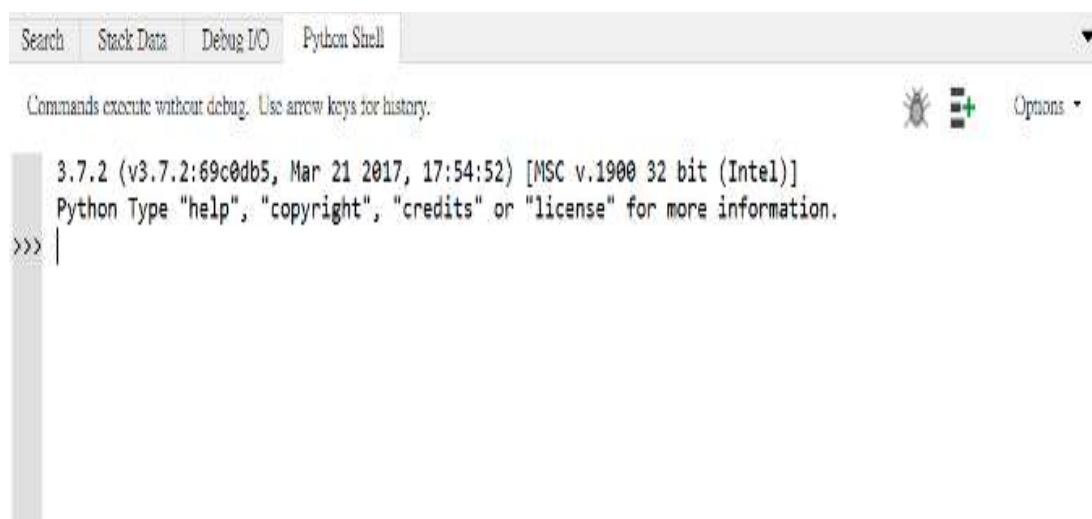


- 7 Follow the steps of the installer to install Wing 101. Make sure to agree to the license agreement to be able to use Wing.

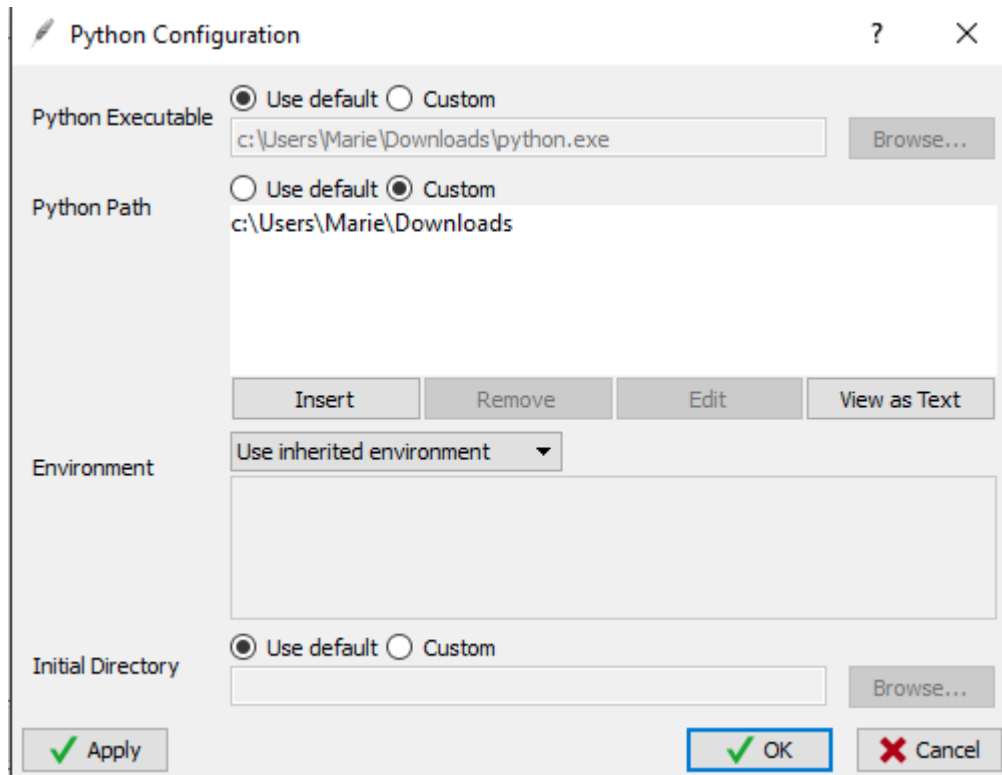
- 8 Once the installation is finished, you might need to lookup Wing 101 in your programs to be able to open it.  
Open up Wing 101, and you should get the "License Agreement" window (first time only). Press "Accept" to enable access to the IDE.



- 9 You will now be guided through the process of configuring Python 3.7.2 on Wing 101 v7. It is possible that Python has already been linked to Wing 101. To check, see what is the message on the bottom right-hand box. If it has a message like the one below, Python is already configured. You can skip ahead to step 12 to test your configuration.



- 10 If you have a different message, Python 3.7.2 is not configured properly. To configure Python 3.7.2, go to Edit -> Configure Python. It will open up a window like this:



The image shows a 'Python Configuration' dialog box with a title bar containing a question mark and a close button. The dialog is organized into sections on the left and configuration options on the right.

- Python Executable:** Features a radio button for 'Use default' (selected) and a radio button for 'Custom'. Below the radio buttons is a text field containing 'c:\Users\Marie\Downloads\python.exe' and a 'Browse...' button.
- Python Path:** Features a radio button for 'Use default' and a radio button for 'Custom' (selected). Below the radio buttons is a large text area containing 'c:\Users\Marie\Downloads'. At the bottom of this section are four buttons: 'Insert', 'Remove', 'Edit', and 'View as Text'.
- Environment:** Features a dropdown menu currently set to 'Use inherited environment' and a large empty text area below it.
- Initial Directory:** Features a radio button for 'Use default' (selected) and a radio button for 'Custom'. Below the radio buttons is a text field and a 'Browse...' button.

At the bottom of the dialog are three buttons: 'Apply' (with a green checkmark icon), 'OK' (with a green checkmark icon and a blue border), and 'Cancel' (with a red X icon).

- 11 Select the custom option and then press on Browse...

The image displays two sequential screenshots of the 'Python Configuration' dialog box. In the top screenshot, the 'Python Executable' field is set to 'c:\Users\Marie\Downloads\python.exe' and the 'Custom' radio button is selected. In the bottom screenshot, the 'Browse...' button next to the 'Python Executable' field is highlighted with a yellow circle.

**Python Configuration**

Python Executable: ☒ Use default ☐ Custom  
c:\Users\Marie\Downloads\python.exe

Python Path: ☐ Use default ☒ Custom  
c:\Users\Marie\Downloads

Environment: Use inherited environment

Initial Directory: ☒ Use default ☐ Custom

**Python Configuration**

Python Executable: ☐ Use default ☒ Custom  
c:\Users\Marie\Downloads\python.exe

Python Path: ☐ Use default ☒ Custom  
c:\Users\Marie\Downloads

Environment: Use inherited environment

Initial Directory: ☒ Use default ☐ Custom

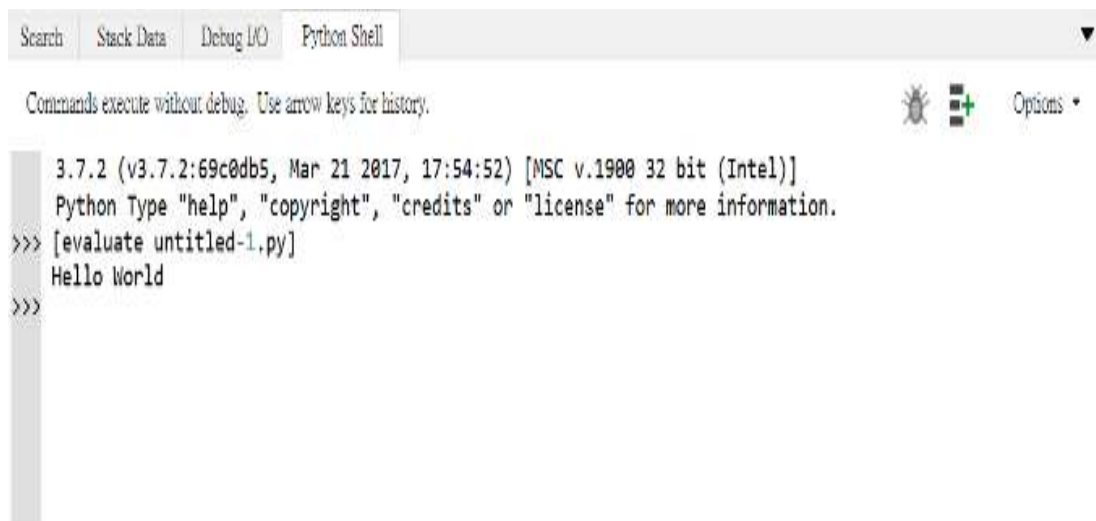
It will open up a window with your files. If you followed the automatic procedure of the Python installation, try typing c:\Users\your-user\AppData\Local\Programs before you press Browse... and you should have a folder called Python where you will find your Python file. If not, navigate to the location where you saved Python and open it.



- 12 When you have found Python, press Ok in the Python Configuration box. You should get a pop-up asking if Wing can restart your Python shell. Press OK.  
To double-check your Python set-up, open a new file and type `print("Hello World")` in the top window. Then, save your file somewhere on your computer! Then press on the green arrow button in the tool bar at the top to run the program. **If you don't save your file, the green arrow button will be gray and you will not be able to press it!!!**




- 13 Once you've pressed Run, the bottom right-hand window should have Hello World in it.



### Instructions for Mac Users

If you are using a Windows operating system, see above.

- 0 Click on the Apple logo on the top right corner , then "About This Mac" to check the version of your operating system. The current version is only compatible with OS X / macOS 10.12+. Please update your mac if you are using a system older than 10.12, to make sure you can use Wing 101 IDE.



- 1 Go to the Python Download page, which is located at <https://www.python.org/downloads>. You should see the following on the top of the page. Python 3 still needs to be installed, even though Python 2 should already be installed.



- 2 Click the button that says **Download Python3.7.2** (or whatever the latest version is). It should immediately start a download of Python. If you are redirected to another page, scroll all the way to the bottom and choose the download option that matches your Mac version.

- 3 Double click the file you have downloaded to install Python 3.7.2. It should open up an installation window like this one:



You can follow the default settings for the installation.

- 4 Now, you will be guided through the Wing IDE v7 installation process. **Wing 101** is an Integrated Development Environment (IDE) like Dr.Racket. It allows you to write and run code in one program. Go to the Wing IDE v7 (or a later version is fine as well) download page, which is located at <http://wingware.com/downloads/wingide-101>. Make sure it says that this is the free version of Wing.

#### Wing 101 - Version 7.0.1 - Released 2019-04-18

Wing 101 is a very simple free Python IDE designed for teaching beginning programmers. It omits most features found in Wing Pro. [Compare Products](#)

If you are new to programming, check out the book [Python Programming Fundamentals](#) and accompanying screen casts, which use Wing IDE 101 to teach programming with Python.

Wing 101 is free to use for any purpose and does not require a license to run.

#### Download Wing 101:

OS X Installer  
64-bit Intel

Wing 101 is a free Python IDE designed for teaching beginning programmers.

[Supported OSes](#)

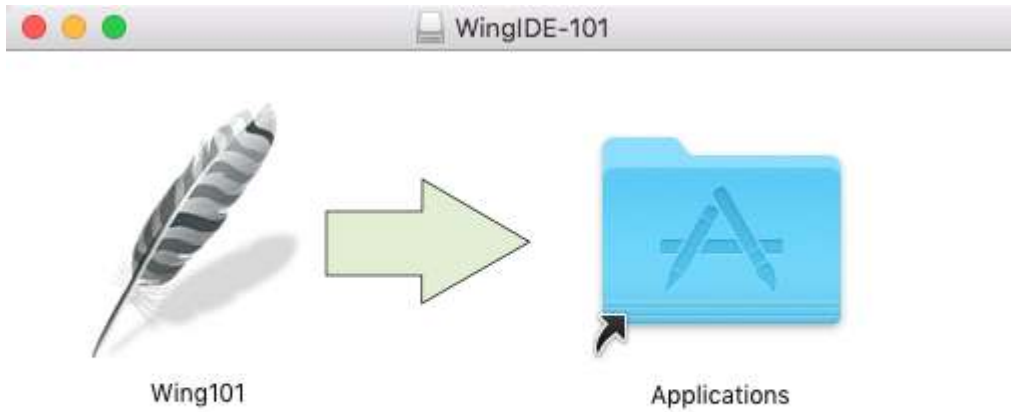
[Supported Python Versions](#)

- 5 Press the green button on the right-hand side of the page to download the installer.

#### Download Wing 101:

OS X Installer  
64-bit Intel

- 6 Double click the DMG file you have downloaded. The following will show up:



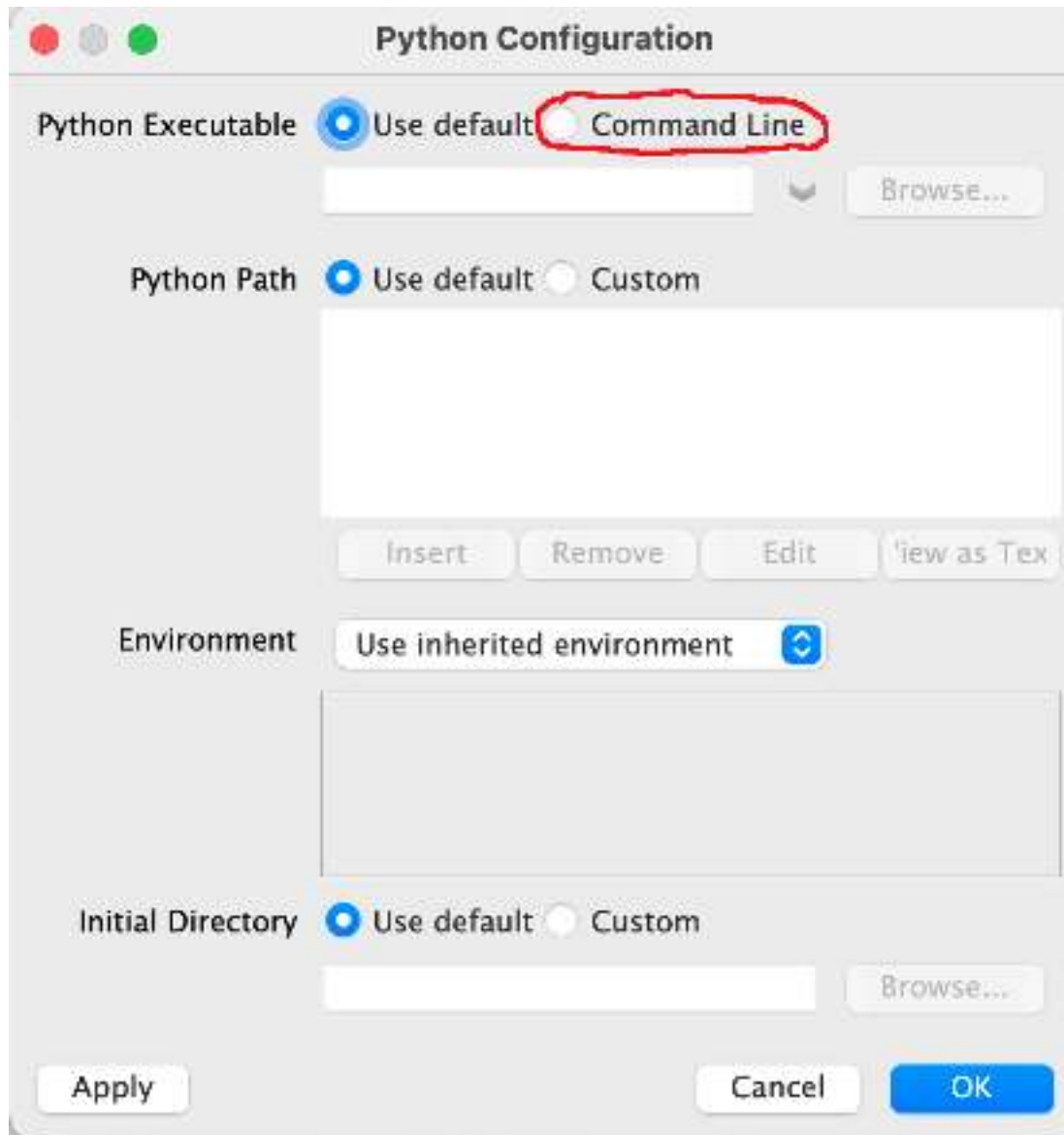
Follow the arrow to drag Wing101 to the **Applications** folder. It might take a few minutes to finish copying.

- 7 Open Wing101 from the **Applications** folder. For the first time you open Wing101, the following warning will be shown:



Click **Open** to open the application and this warning will not show up again. You may also choose **Keep in Dock** for future convenience.

- 8 You will now be guided through the process of configuring Python 3.7.2 on Wing 101 v7. Go to **Edit -> Configure Python**. It will open up a window like this:



Select the **Custom** option circled in red. (Do not click on any other button as you are about to find the location of the Python Executable.)

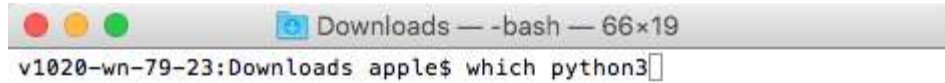
- 9 You will now be guided to find the location of the Python Executable.  
Open up **Terminal** on your mac.



There are two ways to open **Terminal**:

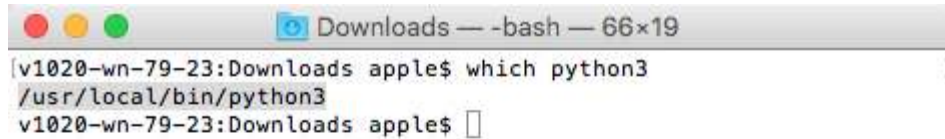
- (1) The location of **Terminal** is usually: **Applications/Utilities/Terminal**. You may find the exact location and open it.
- (2) Using **Spotlight Search** (the magnifying glass icon on the top-right corner of the menu bar) to search for **Terminal** on your Mac and then open it.

- 10 After opening Terminal, type **which python3** on the command line (as following) and then press **Enter**.



```
v1020-wn-79-23:Downloads apple$ which python3
```

The location of Python Executable will be shown on a separate line after pressing Enter, as highlighted in the following image:

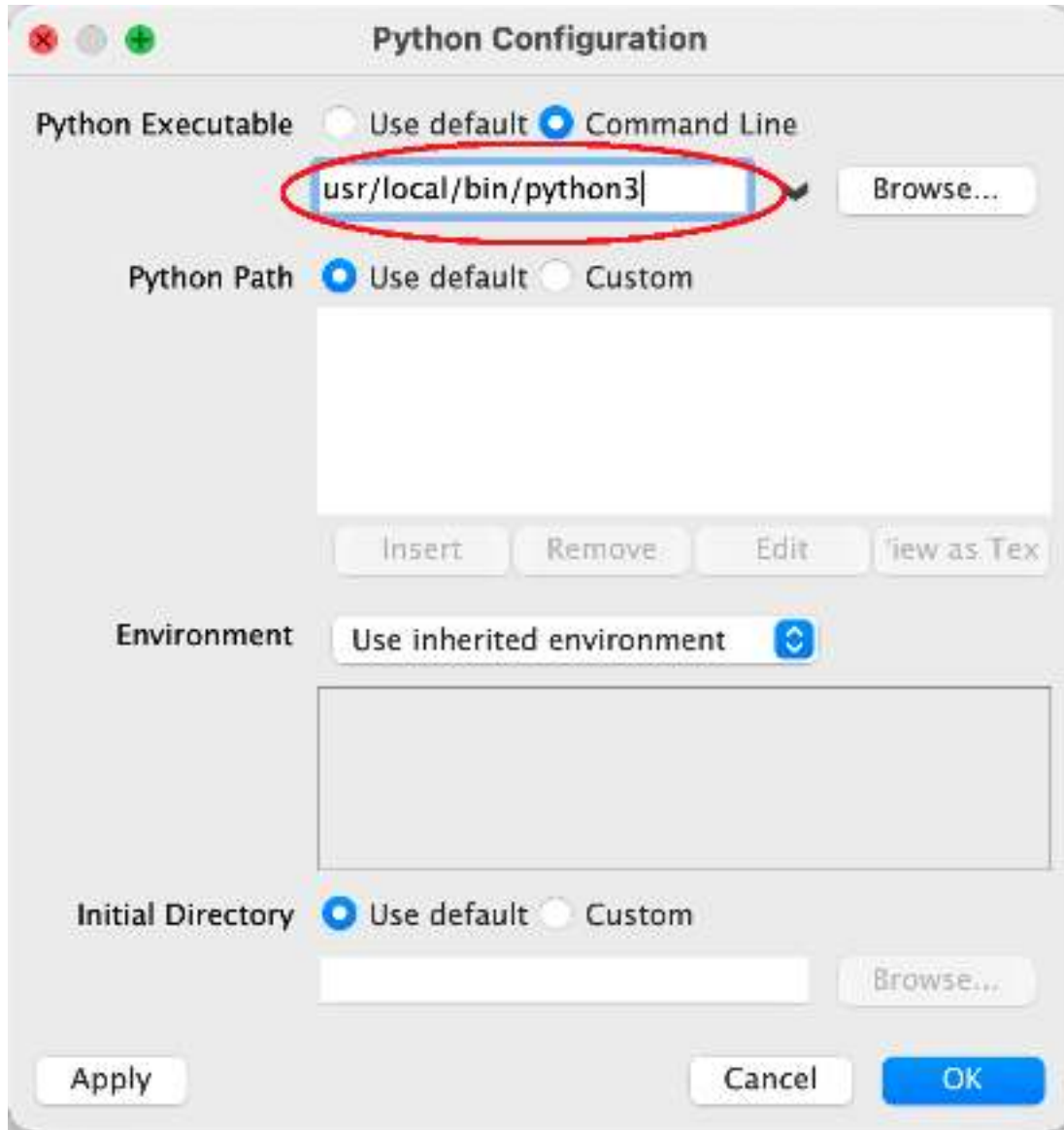


```
v1020-wn-79-23:Downloads apple$ which python3
/usr/local/bin/python3
v1020-wn-79-23:Downloads apple$
```

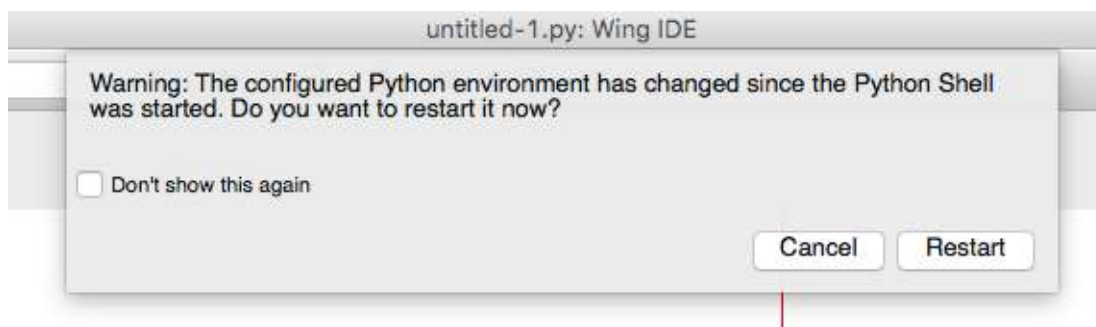
**Copy** the highlighted line.

Note that you may or may not have a different address than the one in the demo image.

- 11 Go back to Wing101 and **paste** the address under Custom option, as circled in red in the following image. Then, click **Apply**.

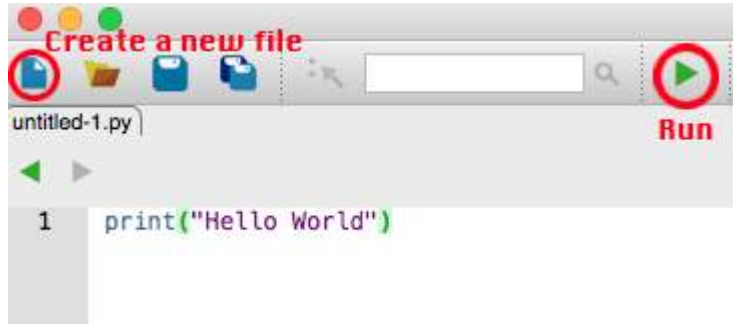


The following warning will show up, then click **Restart**.





- 12 To double-check your Python set-up, open a new file and type `print("Hello World")` in the top window. Then, save your file somewhere on your computer! Then press on the green arrow button in the tool bar at the top to run the program. **If you don't save your file, the green arrow button will be gray and you will not be able to press it!!!**



Hello World should show up in the Python Shell at the bottom.



## Course Staff and Contact Information

### Course Staff and Contact Information

### Technical Difficulties with EdX

If you are experiencing any technical difficulties with viewing content on EdX (for example, videos not loading etc.) please contact [cs-online@uwaterloo.ca](mailto:cs-online@uwaterloo.ca) who can help you debug. Please do not contact this address with course content related issues (post on our forums or ask instructors - see below) or issues with enrolling into courses (contact your academic advisor).

### Online Forum: EdX

In this course, we will be using the embedded online discussion board on EdX. On this site, students will be able to post questions and receive answers to these inquiries. In fact, there is already a post available that will help you with your upcoming Warm-Up Quiz in the discussion forum on the Warm-Up Quiz Page! Anything related to course material or to the assignments should be posted on EdX. If there are issues with EdX, try some of the following methods to contact personnel. Expect replies on the discussion forums within 24 hours from 8:30-5:30 Monday to Friday.

### Announcements: EdX

For announcements, we will use EdX's built in announcement feature. These can be found on the [Updates](#) page. To locate this, go to the main page and click on the right hand side under "Course Tools" and click on "Updates" (see below). Note also, when new announcements are available, a banner will also appear at the top of every page (see below) informing you of a new announcement and give you a quick way to navigate to the page. It is extremely important that you read these announcements and it is your responsibility to make sure you are logging on daily to make sure you're getting new announcements for this course if available.



### Instructors

Instructors are there to help you with the content of the course. The first place to reach them is with discussion forum mentioned above.

We have several instructors this term:



**Mahdiah Ahmadi.** Mahdiah has been working as a postdoctoral fellow at the David R. Cheriton School of Computer Science since 2021. With a Ph.D. in Computer Engineering, she works on making improvements to 5G networks using machine learning techniques in collaboration with Rogers. During her free time, she likes to read about history and stay active through biking and hiking. Email: [m4ahmadi@uwaterloo.ca](mailto:m4ahmadi@uwaterloo.ca)



**Lori Case.** Lori is a lecturer in Computer Science and is the Associate Dean, Co-operative Education, for the Faculty of Mathematics. Lori has taught CS115 and CS116 many times, and enjoys helping students develop a love for Computer Science (or, at least, to become comfortable with the concepts, and be interested in learning more about it). Lori is from St. John's, Newfoundland, and completed her Masters and Ph.D. here at Waterloo. She is an avid knitter, a beginning curler, and a sometimes reluctant player of board games (who almost always enjoys the game, win or lose!). Email: [lori.case@uwaterloo.ca](mailto:lori.case@uwaterloo.ca)



**Gang Lu.** Gang is the Undergraduate Operations Coordinator in School of Computer Science. She is also CS241 ISC for many years. She has MS degree in Computer Science and MS degree in Biochemistry from University of New Hampshire, US. She taught CS136 in Winter 2021 and C116 in Winter 2022. Gang enjoys helping the students to learn. Email: [glu@uwaterloo.ca](mailto:glu@uwaterloo.ca)



**Victoria Sakhnini.** Victoria is a lecturer at the David R. Cheriton School of Computer and is the Associate Director of Software Engineering programs. Victoria has taught CS115/CS116 several times. Victoria loves teaching because contributing to students' intellectual growth is profoundly satisfying. You can find more at her [university](#) and [personal website](#). Victoria can be contacted at [vsakhnini@uwaterloo.ca](mailto:vsakhnini@uwaterloo.ca).

**Shi Zong.** Shi Zong is now a postdoctoral fellow at the David R. Cheriton School of Computer Science. Shi received his Ph.D. degree from The Ohio State University. Shi's research interest is computational linguistics. Email: [shi.zong@uwaterloo.ca](mailto:shi.zong@uwaterloo.ca).

Instructional Support Group (ISG)

In most introductory computer science courses, you will notice that we have a wide range of staff to help support our students.

#### Instructional Support Coordinator (ISC)



Our ISC for this course is **Scott King**. He can be reached at [sfking@uwaterloo.ca](mailto:sfking@uwaterloo.ca). You should contact Scott for any issues regarding illnesses or exceptional situations.

#### Instructional Support Assistants (ISA)

Our course also has several ISAs to help with academic learning. They can be reached at [cs116@uwaterloo.ca](mailto:cs116@uwaterloo.ca) or the preferred method via the EdX discussion forums. Below are the ISAs for this course:



**Anson Huang.** Anson is a 2B Mathematics student pursuing a double major in Combinatorics & Optimization and Computational Mathematics with a specialization in Mathematics/Teaching. Anson spends much of his spare time playing and listening to music and has been in UW Animusic Ensembles since the start of 2022. Apart from music, Anson also likes to improve his skills in archery and volleyball.



**James Plouffe.** James is a 2B undeclared Honours Mathematics student hoping to pursue a minor in French. This is his first term being an ISA. CS 116 is one of his favourite courses and helped him discover his budding passion for computer science. James is a self-proclaimed music nerd and loves listening to, discussing, creating, and reviewing music. James also enjoys reading, creative writing, learning languages, going to concerts, travelling, and spending time outside. He hopes you have as much fun with CS 116 as he did.



**Jonathan Luo.** Jonathan is a 3A Math student following Information Technology Management. This is his first time being an ISA at the University of Waterloo. Jonathan has an interest in creating video games and learning about financial investments. Some things he also does alone in his free time are reading and writing fiction, playing tabletop role-playing games, as well as playing video games.



**Jacob Smith.** Jacob is a 2nd year honours math student. He took CS 116 last winter, was his favourite course and he wants to work on it now. Jacob is a big fan of bowling, hockey, camping, travelling, chess, and gaming. He looks forward to meeting and helping students this term.

### Instructional Assistant (IA)

Our course also has IAs that will help out with office hours and on EdX. Below is the information for our IAs:



**Siqing Huo.** Siqing is currently a MMath-thesis student at the University of Waterloo. She graduated from the University of Waterloo with a Bachelor of Computer Science degree in 2022. Siqing was a Teaching Assistant for this course during the Spring 2022 term and is currently an Instructional Assistant for the Winter 2023 term. In her free time, Siqing enjoys painting.



**Luis Fernando Lins.** Luis is a Master's (Thesis) student at the University of Waterloo, Cheriton School of Computer Science. His research is on Software Engineering, primarily focused on chatbots and business processes. Professionally, Luis has worked as a software developer for over 2 years before becoming a full-time Master's student. In his free time, you will find him writing music or playing the guitar/piano, doing yoga, or hanging out with friends.

**Andreea Pocol.** Andreea is a Ph.D. student at the University of Waterloo in the Cheriton School of Computer Science. Her research area is a cross-section of computer graphics and computer vision, with previous research focusing on stereoscopic 3D and film, and ongoing research including AI text-to-image synthesis and deepfakes. In 2021, she graduated with an Honours Bachelor of Computer Science and a minor in English Language & Literature from the University of Waterloo, with notable co-op work terms at Ubisoft, Oracle, and Christie Digital. She enjoys learning about ancient history, particularly classical antiquity, as well as reading, writing, video editing, photography, playing piano, and illustrating poetry.



**Justin San Juan.** Justin is an MMath Thesis Computer Science student at the University of Waterloo since 2021. He has a dual-degree Bachelor of Engineering in Computer Science and Bachelor of Business Administration in General Business Management from The Hong Kong University of Science and Technology (HKUST) in 2020. Justin is currently working on his thesis on Flashpoint: Reducing the Cost of GPU Cold Starts in Serverless Deep Learning Inference Serving. He is engaged in entrepreneurship, building trading algorithms, and prototyping a personal finance mobile application. In his free time, Justin enjoys reading books on architecture, programming, and entrepreneurship, as well as playing soccer, badminton, and piano.

## Course Description and Learning Outcomes

### Course Description

This course builds on the techniques and patterns learned in CS 115 while making the transition to use of an imperative language. Generative and structural recursion; mutation (assignment) and its role in an imperative language; primitive types and basic I/O; sequencing, selection, and looping; function definition and use; file and console I/O; and issues in computer science will all be examined.

### Learning Outcomes

After completing this course, students will be able to:

- Given a clear and concise statement of a problem or task, write a program from scratch of up to a hundred lines of properly-formatted, tested, and documented Python code to solve the problem or carry out the task
- Use higher-order functions to improve the readability and efficiency of programs
- Use various forms of recursion (structural, accumulative, and generative) in programs
- Use various forms of iteration (for, while) in programs
- Describe the basic memory model for mutation of basic types, lists, and objects in Python
- Distinguish between constant, linear, quadratic, and exponential running times of algorithms
- Explain the relative advantages and disadvantages of lists and dictionaries
- Write useful Python programs using console and file input and output for practical tasks



### Winter: Course Schedule

Modules will be released on a weekly basis. Some modules will take a bit longer and this extra time is built into the structure of the course. The reasoning for the weekly release is that we want to let ideas simmer and for you to remember content from previous modules. Modules will be released on Mondays and students will have approximately one week to go through the content of the module and the assignments.

All times are given in Eastern Time. See <https://24timezones.com/time-zone/et> for more information about Eastern Time and [Ontario, Canada Time Converter](#) for more details on converting.

Module	Release Date	Wrap-Up Quiz Due Date	Assignment Due Date
1	Monday January 9th, 2023 at 12:01am ET	Monday January 23rd, 2023 at 11:59pm ET	Wednesday January 25th, 2023 at 10:00am ET
2	Monday January 16th, 2023 at 12:01am ET	Monday January 30th, 2023 at 11:59pm ET	Wednesday February 1st, 2023 at 10:00am ET
3	Monday January 23rd, 2023 at 12:01am ET	Monday February 6th, 2023 at 11:59pm ET	Wednesday February 8th, 2023 at 10:00am ET
4	Monday January 30th, 2023 at 12:01am ET	Monday February 13th, 2023 at 11:59pm ET	Wednesday February 15th, 2023 at 10:00am ET
5	Monday February 6th, 2023 at 12:01am ET	Monday February 27th, 2023 at 11:59pm ET	Wednesday March 1st, 2023 at 10:00am ET
6	Monday February 13th, 2023 at 12:01am ET	Monday March 13th, 2023 at 11:59pm ET	Wednesday March 15th, 2023 at 10:00am ET
7	Monday February 27th, 2023 at 12:01am ET	Monday March 20th, 2023 at 11:59pm ET	Wednesday March 22nd, 2023 at 10:00am ET
8	Monday March 6th, 2023 at 12:01am ET	Monday March 27th, 2023 at 11:59pm ET	Wednesday March 29th, 2023 at 10:00am ET
9	Monday March 13th, 2023 at 12:01am ET	Monday April 3rd, 2023 at 11:59pm ET	Wednesday April 5th, 2023 at 10:00am ET
10	Monday March 20th, 2023 at 12:01am ET	Monday April 10th, 2023 at 11:59pm ET	Monday, April 10th 2023 at 11:59PM ET

### Detailed Daily Course Schedule



In this section we give an outline of approximately what we'll be covering on each day. If you wanted to follow along online, you can do that as well and we give our recommendations for what content will be relevant and when to cover it. Please note that unexpected closures such as those relating to snow days might alter the schedule below.

<b>Date</b>	<b>Relevant Online Course content</b>	<b>Deadlines/Class Work</b>
<b>M:</b> January 9	Orient Yourself with the Course, Style Guide, Syllabus Sections, Read Assignment 1, Warm-Up Quiz 1	
<b>T:</b> January 10	Lesson 1.1, Lesson 1.2, Lesson 1.3	Slides up to Module 1 slide 20
<b>W:</b> January 11	Lesson 1.4, Lesson 1.5, M1 Wrap-Up Quiz	
<b>R:</b> January 12	Start Assignment 1	Slides up to Module 1 slide 40
<b>F:</b> January 13	Continue Assignment 1	Tutorial covering course basics and Warm-up Quiz 1
<b>M:</b> January 16	Lesson 2.1, Lesson 2.2	
<b>T:</b> January 17	Continue Assignment 1, Read Assignment 2	Slides up to Module 2 Slide 7
<b>W:</b> January 18	Lesson 2.3, Lesson 2.4, M2 Wrap-Up Quiz	
<b>R:</b> January 19	Start Assignment 2	Slides up to Module 2 Slide 17
<b>F:</b> January 20	Continue Assignment 2	Tutorial covering Wrap-Up Quiz 1
<b>M:</b> January 23	Lesson 3.1, Continue Assignment 2	<b>Warm up Quiz 1, Wrap-Up Quiz 1 due at 11:59pm</b>
<b>T:</b> January 24	Lesson 3.2	Slides up to the end of Module 2
<b>W:</b> January 25	Lesson 3.3	<b>Assignment 1 Due at 10:00am ET</b>
<b>R:</b> January 26	Lesson 3.4	Slides up to Module 3 Slide 20
<b>F:</b> January 27	Lesson 3.5 and M3 Wrap-Up Quiz	Tutorial covering Wrap-Up Quiz 2
<b>M:</b> January 30	Lesson 4.1, Read Assignment 4	<b>Wrap-Up Quiz 2 due at 11:59pm</b>

<b>Date</b>	<b>Relevant Online Course content</b>	<b>Deadlines/Class Work</b>
<b>T:</b> January 31	Lesson 4.2	Sildes up to the end of Module 3
<b>W:</b> February 1	Lesson 4.3	<b>Assignment 2 Due at 10:00am ET</b>
<b>R:</b> February 2	Lesson 4.4	Sildes up to Module 4 slide 24
<b>F:</b> February 3	Lesson 4.5	Tutorial covering Wrap-Up Quiz 3
<b>M:</b> February 6	Lesson 4.6	<b>Wrap-Up Quiz 3 due at 11:59pm</b>
<b>T:</b> February 7	Lesson 4.7	Sildes up to Module 4 slide 35
<b>W:</b> February 8	M4 Wrap-Up Quiz, Start Assignment 4	<b>Assignment 3 Due at 10:00am ET</b>
<b>R:</b> February 9	Continue Assignment 4	Sildes up to Module 5 slide 5
<b>F:</b> February 10	Continue Assignment 4	Tutorial covering Wrap-Up Quiz 4
<b>M:</b> February 13	Lesson 5.1, Read Assignment 5	<b>Wrap-Up Quiz 4 due at 11:59pm</b>
<b>T:</b> February 14	Lesson 5.2	Sildes up to Module 5 slide 29
<b>W:</b> February 15	Lesson 5.3	<b>Assignment 4 Due at 10:00am ET</b>
<b>R:</b> February 16	M5 Wrap-Up Quiz, Start Assignment 5	Sildes up to Module 5 slide 44
<b>F:</b> February 17	Continue Assignment 5	Tutorial covering Wrap-Up Quiz 5
February 20-24	Reading Week. Use the time wisely!	
<b>M:</b> February 27	Lesson 6.1, Read Assignment 6	<b>Wrap-Up Quiz 5 due at 11:59pm</b>
<b>T:</b> February 28	Lesson 6.3, M6 Wrap-Up Quiz	Sildes up to Module 6 slide 13
<b>W:</b> March 1	Study for Midterm	<b>Assignment 5 Due at 10:00am ET</b>
<b>R:</b> March 2	Study for Midterm	Sildes up to the end of Module 6
<b>F:</b> March 3	Study for Midterm	Tutorial covering Midterm Preparation
<b>M:</b> March 6	Study for Midterm	<b>Midterm Exam 7:00pm</b>

<b>Date</b>	<b>Relevant Online Course content</b>	<b>Deadlines/Class Work</b>
<b>T:</b> March 7	Start Assignment 6	Sildes up to Module 7 slide 12
<b>W:</b> March 8	Continue Assignment 6	
<b>R:</b> March 9	Continue Assignment 6	Sildes up to Module 7 slide 26
<b>F:</b> March 10	Lesson 7.1, Read Assignment 7	Tutorial covering Wrap-Up Quiz 6
<b>M:</b> March 13	Lesson 7.2, Lesson 7.3	<b>Wrap-Up Quiz 6 due at 11:59pm</b>
<b>T:</b> March 14	Lesson 7.4	Sildes up to the end of Module 7
<b>W:</b> March 15	Lesson 7.5, M7 Wrap-Up Quiz	<b>Assignment 6 Due at 10:00am ET</b>
<b>R:</b> March 16	Start Assignment 7	Sildes up to Module 8 slide 22
<b>F:</b> March 17	M8 Warm-Up Quiz, Continue Assignment 7	Tutorial covering Wrap-Up Quiz 7
<b>M:</b> March 20	Lesson 8.1, Lesson 8.2, Read Assignment 8	<b>Wrap-Up Quiz 7 due at 11:59pm</b>
<b>T:</b> March 21	Lesson 8.3, Lesson 8.4	Sildes up to Module 9 slide 6
<b>W:</b> March 22	Lesson 8.5, Lesson 8.6, M8 Wrap-Up Quiz	<b>Assignment 7 Due at 10:00am ET</b>
<b>R:</b> March 23	Start Assignment 8	Sildes up to Module 9 slide 28
<b>F:</b> March 24	M9 Warm-Up Quiz, Continue Assignment 8	Tutorial covering Wrap-Up Quiz 8
<b>M:</b> March 27	Lesson 9.1, Read Assignment 9	<b>Wrap-Up Quiz 8 due at 11:59pm</b>
<b>T:</b> March 28	Lesson 9.2	Sildes up to Module 10 slide 9
<b>W:</b> March 29	Lesson 9.3, M9 Wrap-Up Quiz	<b>Assignment 8 Due at 10:00am ET</b>
<b>R:</b> March 30	Start Assignment 9	Sildes up to Module 10 slide 33
<b>F:</b> March 31	M10 Warm-Up Quiz, Continue Assignment 9	Tutorial covering Wrap-Up Quiz 9

<b>Date</b>	<b>Relevant Online Course content</b>	<b>Deadlines/Class Work</b>
<b>M:</b> April 3	Lesson 10.1, Lesson 10.2, Lesson 10.3	<b>Wrap-Up Quiz 9 due at 11:59pm</b>
<b>T:</b> April 4	M10 Wrap-Up Quiz	Sildes up to end of Module 10
<b>W:</b> April 5	Start Final Exam Preparation	<b>Assignment 9 Due at 10:00am ET</b>
<b>R:</b> April 6	Continue Final Exam Preparation	Review
<b>F:</b> April 7	Good Friday Classes Cancelled	
<b>M:</b> April 10	Continue Final Exam Preparation	Tutorial covering Wrap-Up Quiz 10 <b>Wrap-Up Quiz 10 due at 11:59pm</b>

UW: CS 116

Introduction to Computer Science 2

---

## Winter: Slides

In what follows you can find all the course note pdf slides.

All the files below in a single zip file can be found [here](#).

Slides only:

- [Module 01](#)
- [Module 02](#)
- [Module 03](#)
- [Module 04](#)
- [Module 05](#)
- [Module 06](#)
- [Module 07](#)
- [Module 08](#)
- [Module 09](#)
- [Module 10](#)
- [Module 11](#)
- [Module 12](#)

Three slides per pages with space for notes:

- [Module 01](#)
- [Module 02](#)
- [Module 03](#)
- [Module 04](#)
- [Module 05](#)
- [Module 06](#)
- [Module 07](#)
- [Module 08](#)
- [Module 09](#)
- [Module 10](#)
- [Module 11](#)
- [Module 12](#)

**Winter: Grade Breakdown and Course Policies on Missed Assignments**

In the event that an in person final exam can be written, which is the default assumption for the course, we have the following grading scheme:

Component	Weight (%)
Warm-Up Quiz 1 and Wrap-Up Quizzes	5%
Assignments	20%
Midterm Exam	30%
Final Exam	45%

**Important!** You **must** pass the weighted exam average (at least 50%) of the midterm exam and the final exam to pass the course. Failure to pass the weighted exam average will result in a final grade of either 46% or your weighted exam average, whichever is lower.

**Quiz Questions**

The quizzes will each have one attempt for non-programming problems and unlimited attempts for programming problems. The quizzes will be weighted equally (not according to how many problems solved per quiz as some quizzes might have more problems than others) so each quiz is worth 0.5% of your final grade. We will take your best 10 of 11, including the first warm-up quiz which is an orientation quiz to help you get acquainted with the website.

**Concept Checks**

Please note that concept checks are included here only for practice purposes. They will not be graded in any way. However, students are free to ask questions on discussion forums and in office hours.

Solutions to concept checks **will not be provided to students**. There are several reasons for this:

1. We often don't have solutions to the problems. (Most of our test cases are done using the check module).
2. Out of concern for academic integrity this term and across different terms, we do not feel we should release either test data or solutions.
3. With respect to the above, more is learnt when you **struggle to find the answer** than is learnt when you quickly read a solution. In other words, active learning is far better than passive learning. Note that we will still post solutions to all assignments so you will get solutions to some problems. However it cannot be stressed enough that struggling to find the answer is worth more than being given it. This being said, staff is more than happy to help students get to the answer via office hours either before or after the due

dates.

In order to maintain a level of success in the course, students should not miss the completion of any modules.

### Progress

There is a progress tab that you can find near the top left of this page to let you know how you are doing in the quizzes and Concept Check questions.

### Assignments

Assignments will be submitted to MarkUs through our online platform. The assignment that is marked is the last on time submitted version per question [on time according to the time on MarkUs which sometimes allows for a few minutes of grace period near the deadline]. Once you submit an assignment to MarkUs, you will receive an email consisting of basic tests that you passed or failed. Students should check their basic tests email to ensure that the code meets the specification exactly. We will not accept submissions that do not match our test output exactly.

Assignments are weighted evenly.

We will also only take your best 9 of the 10 assignments so you can miss a week if your workload becomes too onerous. However, **if you join the course late and cannot complete Assignment 1 on time, this assignment will still count as your lowest assignment grade!** In other words, you do not get to drop two assignments if you join after the deadline of Assignment 1.

Remark requests for assignments can be made up to one week after the assignment has been returned.

### Missed Assessments

**There will be no extensions on assignments and late assignments will not be accepted!** Make sure you understand what time zone you are in and when our due dates are in Eastern Time.

If sick, please email our ISC to request an exception after submitting a Verification of Illness form or completing a self-declaration in Quest. Reweighting of assignments is not automatic even with a valid doctor's note and is up to the discretion of the ISC and instructor to allow for reweighting. We will not typically exclude more than 3 assignment components. If you find yourself in this situation, please contact our ISC and your instruction to discuss options.

### Saving Work

When you press on the **Run Code** button or the **Submit Code** button, your work will save. Please make sure you do this often! There is often no recourse if you fail to save your work so please run your code often!

### MOSS

MOSS (Measure of Software Similarities) is used in this course as a mean of comparing students' assignments in order to support academic integrity.

### Academic Integrity

Please note that if you do get caught in an academic integrity offense - the assignment you cheated on cannot be the dropped assignment.



### **Late Adds**

This course does not support students adding after the two week add period as stipulated in the University Calendar under no exceptions. It is too difficult to miss one sixth of this course and catch up meaningfully.

### University Policies and Academic Integrity Submission Times

Please be aware that the University of Waterloo is located in the **Eastern Time Zone** (GMT or UTC-5 during standard time and UTC-4 during daylight saving time) and, as such, the time that your activities and/or assignments are due is based on this zone. If you are outside the Eastern Time Zone and require assistance with converting your time, please try the [Ontario, Canada Time Converter](#).

### Accommodation Due to Illness

**If your instructor has provided specific procedures for you to follow if you miss assignment due dates, term tests, or a final examination, adhere to those instructions.** Otherwise:

### Missed Assignments/Tests/Quizzes

Contact the ISC, **not your instructor**, as soon as you realize there will be a problem, and preferably within 48 hours, but no more than 72 hours, have a medical practitioner complete a [Verification of Illness Form](#).

**Email** a scanned copy of the Verification of Illness Form to the ISC of this course. In your email to the ISC, provide your name, student ID number, and exactly what course activity you missed.

Further information regarding Management of Requests for Accommodation Due to Illness can be found on the [Accommodation due to illness](#) page.

### Missed Final Examinations

If this course has a final exam and if you are unable to write a final examination due to illness, seek medical treatment and have a medical practitioner complete a [Verification of Illness Form](#). Email a scanned copy to the Centre for Extended Learning (CEL) at [extendedlearning@uwaterloo.ca](mailto:extendedlearning@uwaterloo.ca) within 48 hours of your missed exam. Make sure you include your name, student ID number, and the exam(s) missed. You will be REQUIRED to hand in the original completed form before you write the make-up examination.

After your completed Verification of Illness Form has been received and processed, you will be emailed your alternate exam date and time. This can take up to 2 business days. If you are within **150 km** of Waterloo you should be prepared to write in Waterloo on the additional CEL [exam dates](#). If you live outside the 150 km radius, CEL will work with you to make suitable arrangements.

Further information about [Examination Accommodation Due to Illness](#) regulations is available in the Undergraduate Calendar.

### Academic Integrity

Before beginning, we strongly encourage students to check out <https://uwaterloo.ca/academic-integrity>.

[/academic-integrity-modules-students-stem](#) and to complete these modules. They should take an hour total but the time invested in understanding university academic integrity policies could save you a lot of heartache down the road. What follows is a brief summary and addendum to the content on the aforementioned site.

- In order to maintain a culture of academic integrity, members of the University of Waterloo community are expected to promote honesty, trust, fairness, respect, and responsibility. **If you have not already completed the online tutorial regarding academic integrity you should do so as soon as possible.** Undergraduate students should see the [Academic Integrity Tutorial](#) and graduate students should see the [Graduate Students and Academic Integrity](#) website.
- Proper citations are part of academic integrity. Reminder that **you are not permitted to look up either full or partial solutions online or in printed sources.** If using course material in your solutions, simply cite where it came from in a comment. There is no formal style here so long as it is easy to understand.
- **Do not post your code or any questions from this course anywhere publicly online.** This includes but is not limited to on Github, StackOverflow, Reddit, CourseHero, Chegg and so on. You may create private repositories however to share with employers if you deem this necessary. Violators will be treated as academic integrity offenses and can carry penalties including -5% on your course grade. This penalty can even be applied for cheating that occurs on Concept Check and Quiz questions! Please refer to [The University of Waterloo's Code Sharing Policy](#) for more information.
- Each assignment question will require you to digitally sign an academic integrity statement before you can view the questions. Failure to accept the statement will result in you not being able to view the assignments and hence result in your assignments and projects not being graded.
- Please note that if you do get caught in an academic integrity offense - the assignment you cheated on cannot be the dropped assignment.
- Please note that we encourage high level discussion of your assignments but do not share your assignment, code or precise details with anyone else in or outside this course (with the exception being instructors or ISAs/IAs in the course). When you write your final version of your code, you should do so without your notes to ensure that you've understood the material. We recommend posting on our discussion forum for help and framing your question in a way that it can be made public. Code posts on our discussion forum should **always** be private.
- **MOSS (Measure of Software Similarities)** is used in this course as a mean of comparing students' assignments in order to support academic integrity.
- The use of Copilot or similar products is expressly prohibited and will result in 0 on assignments, a deduction on final course grade and further penalties as specified by the Associate Dean.
- For further information on academic integrity, please visit the [Office of Academic Integrity](#).

## Discipline

A student is expected to know what constitutes [academic integrity](#) to avoid committing an academic offence, and to take responsibility for his/her actions. A student who is unsure whether an action constitutes an offence, or who needs help in learning how to avoid offences (e.g., plagiarism, cheating) or about "rules" for group work/collaboration, should seek guidance from the course instructor, academic advisor, or the undergraduate Associate Dean. For information on categories of offences and types of penalties, students should refer to [Policy 71 - Student Discipline](#). For typical penalties, check [Guidelines for the Assessment of Penalties](#).

## Appeals

A decision made or penalty imposed under [Policy 70 - Student Petitions and Grievances](#), (other than a petition) or [Policy 71 - Student Discipline](#), may be appealed if there is a ground. A student who believes

he/she has a ground for an appeal should refer to [Policy 72 - Student Appeals](#).

### **Grievance**

A student who believes that a decision affecting some aspect of his/her university life has been unfair or unreasonable may have grounds for initiating a grievance. Read [Policy 70 - Student Petitions and Grievances](#), Section 4. When in doubt please be certain to contact the department's administrative assistant who will provide further assistance.

### **Final Grades**

In accordance with [Policy 46 - Information Management](#), Appendix A - Access to and Release of Student Information, the Centre for Extended Learning does not release final examination grades or final course grades to students. Students must go to [Quest](#) to see all final grades. Any grades posted in Waterloo LEARN are unofficial.

### **AccessAbility Services**

[AccessAbility Services](#), located in Needles Hall, collaborates with all academic departments to arrange appropriate accommodations for students with disabilities without compromising the academic integrity of the curriculum. If you require academic accommodation to lessen the impact of your disability, please register with AccessAbility Services at the beginning of each academic term and for each course.

### **Accessibility Statement**

The Centre for Extended Learning strives to meet the needs of all our online learners. Our ongoing efforts to become aligned with the [Accessibility for Ontarians with Disabilities Act \(AODA\)](#) are guided by University of Waterloo accessibility [Legislation](#) and policy and the [World Wide Web Consortium's \(W3C\) Web Content Accessibility Guidelines \(WCAG\) 2.0](#). The majority of our online courses are currently delivered via the Desire2Learn Learning Environment. Learn more about [Desire2Learn's Accessibility Standards Compliance](#).

### **Use of Computing and Network Resources**

Please see the [Guidelines on Use of Waterloo Computing and Network Resources](#).

### **Mental Health Support:**

The Faculty of Math encourages students to seek out mental health support if needed.

#### **On-campus Resources:**

- Campus Wellness <https://uwaterloo.ca/campus-wellness/>
- Counselling Services: [counselling.services@uwaterloo.ca](mailto:counselling.services@uwaterloo.ca)/ 519-888-4567 ext 32655
- MATES: one-to-one peer support program offered by Federation of Students (FEDS) and Counselling Services: [mates@uwaterloo.ca](mailto:mates@uwaterloo.ca)
- Health Services: located across the creek from the Student Life Centre, 519-888-4096.

#### **Off-campus Resources:**

- Good2Talk (24/7): Free confidential help line for post-secondary students. Phone: 1-866-925-5454
- Here 24/7: Mental Health and Crisis Service Team. Phone: 1-844-437-3247
- OK2BME: set of support services for lesbian, gay, bisexual, transgender or questioning teens in Waterloo. Phone: 519-884-0000 extension 213

**Diversity:**

It is our intent that students from all diverse backgrounds and perspectives be well served by this course, and that students' learning needs be addressed both in and out of class. We recognize the immense value of the diversity in identities, perspectives, and contributions that students bring, and the benefit it has on our educational environment. Your suggestions are encouraged and appreciated. Please let us know ways to improve the effectiveness of the course for you personally or for other students or student groups. In particular:

- We will gladly honour your request to address you by an alternate/preferred name or gender pronoun. Please advise us of this preference early in the semester so we may make appropriate changes to our records.
- We will honour your religious holidays and celebrations. Please inform of us these at the start of the course.
- We will follow AccessAbility Services guidelines and protocols on how to best support students with different learning needs.

**Copyright Information****UWaterloo's Web Pages**

All rights, including copyright, images, slides, audio, and video components, of the content of this course are owned by the course author and the University of Waterloo, unless otherwise stated. By accessing this course, you agree that you may only download the content for your own personal, non-commercial use. You are not permitted to copy, broadcast, download, store (in any medium), transmit, show or play in public, adapt, or change in any way the content of these web pages for any other purpose whatsoever without the prior written permission of the course author and the University of Waterloo, Centre for Extended Learning.

**Other Sources**

Respect the copyright of others and abide by all copyright notices and regulations when using the computing facilities provided for your course of study by the University of Waterloo. No material on the Internet or World Wide Web may be reproduced or distributed in any material form or in any medium, without permission from copyright holders or their assignees. To support your course of study, the University of Waterloo has provided hypertext links to relevant websites, resources, and services on the web. These resources must be used in accordance with any registration requirements or conditions which may be specified. You must be aware that in providing such hypertext links, the University of Waterloo has not authorized any acts (including reproduction or distribution) which, if undertaken without permission of copyright owners or their assignees, may be infringement of copyright. Permission for such acts can only be granted by copyright owners or their assignees.

If there are any questions about this notice, please contact the University of Waterloo, Centre for Extended Learning, Waterloo, Ontario, Canada, N2L 3G1 or [extendedlearning@uwaterloo.ca](mailto:extendedlearning@uwaterloo.ca).

**Integrity Statement**

What follows is the official integrity statement you must agree to before each assignment:

I declare the following statements to be true:

- The work I submit here is entirely my own.

- I have not and will not share any of my code with anyone at any point. It is my responsibility to ensure my code is not shared.
- I have not discussed and will not post the contents of this assessment nor its solutions on any public or private forum or website. It is my responsibility to ensure my assessment is not shared.
- Do not look up either full or partial solutions on the internet or in printed resources.
- I am aware that misconduct related to assessments can result in significant penalties, possibly including failure in the course and suspension (this is covered in Policy 71: <https://uwaterloo.ca/secretariat/policies-procedures-guidelines/policy-71>).

Failure to accept the integrity policy will result in your assignment not being graded.

By clicking the box below, it will function as my signature that I have read and understood the above.

**Credits and Copyright**

© Course Author(s) and University of Waterloo.

For further information, please contact the Centre for Extended Learning, University of Waterloo, 200 University Ave. W., Waterloo, ON, Canada, N2L 3G1, [extendedlearning@uwaterloo.ca](mailto:extendedlearning@uwaterloo.ca).

## MarkUs

### MarkUs Integration

Assignments are submitted using MarkUs (see below for login instructions) which can be found here <https://student.cs.uwaterloo.ca/~markus/>. We have integrated this directly into the edX module for seamless integration. However, should something go awry or if you want to work offline, you may follow the instructions below.

Further, it is your responsibility to check the basic tests and to make sure you use the submit code button on each and every individual assignment question. Failing to **submit** code will result in your most recent version of my code to not be submitted on MarkUs and hence not graded. It is **your responsibility** to make sure that you submit my latest version of code and failing to do this will result in your assignment not being graded with no option for recourse.

You may submit and resubmit assignment questions as many times as you would like before the due date. This means it is in your best interest to start early and submit often!

### MarkUs How-To

You can also look at your marks and graded assignment on MarkUs. You can watch how to submit file on MarkUs by clicking on the video below. Every time that the video shows CS 115, make sure you switch it to CS 116.

### Logging into MarkUs

To log in to MarkUs, please ensure that you are doing the following:

- Use Firefox or Chrome to access Markus. MarkUs does *not* work properly with Internet Explorer.
- Use all lowercase letters when entering your userid on the CAS page. Using uppercase letters will render you unrecognizable to the system.

If you encounter a "Login Failed" message, check that you meet the above three criteria before contacting course personnel. If you are still experiencing issues or have run into different errors (such as a redirect problem), please e-mail the course account with your Quest userid and a brief description of your error.

### Submitting Assignments

To submit files for Assignments, follow the instructions below.

1. On your MarkUs homepage, click the assignment name to go to its submission page.


#### Submissions

This box shows you how many files you have submitted and how many of the required files you are missing. The Missing Required Files will initially be equal to the number of files you have to submit. As you submit the properly named files, this number will decrease.



## Assignment Rules

This box has the assignment name, due date, and the names of the required files for the assignment. You must name your assignment submissions as indicated by the list: a lowercase a followed by the assignment number, then a lowercase q followed by the question number. The file extension must be .py. For example, for Assignment 02 Question 3, your file should be named a02q3.py

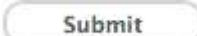
2. Click the "Submissions" tab at the top of the page.
3. Click "Add A New File". A new row will be added to the table. Click "Choose File" or "Submit" in the new row. (The button's name will depend on your web browser.) In the window that appears, browse to where you have saved your file and select it. Click the  button.
4. Your submitted file will now appear in the table. Check that the filename is correct. If you click on the filename, you will see the contents of the file. You should check that the contents are correct, and that you have submitted what you wanted to submit.

It is best to submit each file individually (add a single file and submit, then repeat for all the files you have). While you can submit all your files at once, there have been instances in the past where the files were not properly submitted.

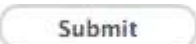
## Replacing a file

You can only replace a file with one that has the same name.

**Warning** Do not use Internet Explorer to replace files. It will appear as though the change has gone through when it has not. Your files will not be replaced.

- Under the "Replace" column, click "Choose File" or "Browse" for the file you want to replace. In the window that appears, browse to the replacement file and select it. Click the  button.

## Deleting a file

If you have submitted the incorrect file, you can delete it by selecting the "Delete" box for the file and then clicking . This is also an alternative way to replace a file: delete the file you want to replace and submit the correct version/file.

## Viewing Assignment Marks and Results

Once marking for an assignment is completed, you can see your mark for each assignment on your MarkUs home page. For a more detailed breakdown of your mark along with marker comments, click the "Results" link.

- On the right is your mark breakdown:

### Marks

Under this tab you can see the level you received for each criterion.

### Summary

Under this tab you can see the weight for each category and how your mark was added up.

- On the left is the code for one of your files.
  1. To see marker comments, click on the left drop-down box (it is above and to the right of "Annot. Summary").
  2. Select "GRADED\_ASSIGNMENT.py". This file contains all of your submissions.
  3. Put your mouse over any lines that are yellow. A box will appear with the marker's comments about that particular line of code, or a general note about your submission.

- The "Annot. Summary" tab contains a list of all the comments a marker has made to your assignment. Click on the link at the top left of each comment to go to the associated code.

**Style Guide****Style Guide**

In this course, we will adhere to the Style Guide. These will be the source for questions regarding documentation concerns for assignments.

**Assignment Solutions**

Click on an assignment name below to download the solutions.

- No solutions have been uploaded for this course.

### Office Hours

Some office hours are held online, as indicated by (Online) in the chart. Some are held in person. For those, the room number will follow the person's name in the table.

**Updates under Course Tools here on EdX is currently the go-to location for information about temporary changes to office hours.**

Office hours are **always busiest just before the assignment is due**. You will save yourself time and frustration by starting your assignment early and attending office hours, if necessary, well before the assignment is due.

If you have any questions please email us at [cs116@uwaterloo.ca](mailto:cs116@uwaterloo.ca) or ask a question on our discussion forum (you can find more information on this under Course Staff and Contact Information). As well, you can email us to set up an appointment if the times below do not work.

**Office hours will begin Tuesday, January 10.**

### Timetable

	Monday	Tuesday	Wednesday	Thursday
8:00 AM				
8:30 AM			Jacob MC4065 8:30 - 10:00	
9:00 AM				
9:30 AM	Lori Online 9:30 - 11:00	Jacob MC4065 9:30 - 11:00		
10:00 AM				
10:30 AM				

11:00 AM				Justin Online 11:00 - 1:00				
11:30 AM								
12:00 PM					Siqing Online 12:00 - 2:00			
12:30 PM								
1:00 PM	Luis MC4065 1:00 - 2:00		Andreea MC4065 1:00 - 5:00				Jonathan MC4065 1:00 - 4:00	
1:30 PM				James MC4065 1:30 - 3:00		James MC4065 1:30 - 3:00		James MC4065 1:30 - 3:00
2:00 PM	Anson MC4065 2:00 - 4:00				Lori Case Online 2:00 - 3:30	Anson Online 2:00 - 4:00		
2:30 PM								
3:00 PM		Siqing Online 3:00 - 5:00						
3:30 PM								Luis Online 3:30 - 5:00
4:00 PM				Gang Online 4:00 - 5:30				
4:30 PM	Victoria DC3113 4:30 - 6:00				Mahdieh Online 4:30 - 6:00		Luis MC4065 4:30 - 6:00	
5:00 PM						Shi Online 5:00 - 6:30		
5:30 PM								
6:00 PM								

#### Instructions for Online office hours

You can use Teams through your browser at [teams.microsoft.com](https://teams.microsoft.com) or you can download the app from [Microsoft's download page](#).

When launching Teams, you will be prompted to sign in. Enter your @uwaterloo.ca address. This should

take you to the UWaterloo sign-in page (just like you saw when signing into to EdX).

Once you are in Teams, you need to select the "Teams" section:



This should take you to a list of Teams you're part of. Select the CS116 team for this term. It's called "CS116 (1205) (Students)". It looks like this (this picture was taken from the 1205 term):



You will see a list of channels, similar to the list shown below.



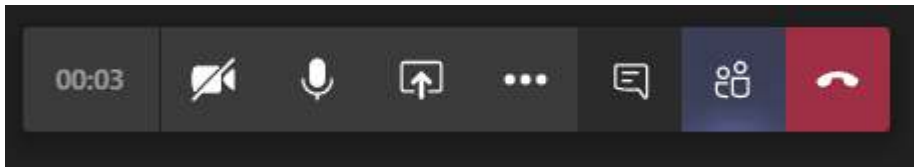
We will have one general chat channel where everyone will congregate to get their questions answered.

Each day, we will post a message for that day's online office hours. When you have a question, please reply to that message, and course staff will call you back. We will also mark your message with a thumb's up.



The Thumbs up helps both instructors and students keep track of where we are in the office hour queue. It is a video call, but you do not need to use a webcam. We are using video calls so that you can show us your screen. If you do not have a microphone, you can use the text chat during the call.

During the call you will see the interface bar. It hides itself automatically, so you may need to wiggle your mouse.



From left to right, the important buttons are "Camera On/Off", "Microphone On/Off", "Share Screen", "...", and "Chat Window". (Depending on your screen size, chat window and/or share screen might get moved into the "..." menu).

To share your screen, click the "Share Screen" button. It will give you the choice of sharing your entire screen, or a single window.



You likely want to share your browser window so that we can see the code you're working on. Please keep in mind that you are showing that window to us. You might want to close any browser tabs you don't want visible. Just saying.

If office hours are busy, we may impose a time limit on your call (you can always go to the back of the line).



**Assignment Regrade Request**

When a regrading is requested, we may reexamine questions other than those reported on this form to ensure that there are no errors in grading. Grading errors may result in the raising or lowering of your grade, or no change to your grade at all.

Which assignment would you like to have regraded?

Question

Describe the perceived grading error

Add Another Question

Submit

UW: CS 116

Introduction to Computer Science 2

---

### VPN Information

Here is a link to accessing our VPN service if you are in a foreign country

<https://wiki.uwaterloo.ca/display/ISTKB/Accessing+Waterloo+learning+technologies+from+China>

UW: CS 116

Introduction to Computer Science 2

---

### Sandbox

Use this sandbox to write Python code...

```
1 print("ASDF")
```

Code Output