

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.

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.

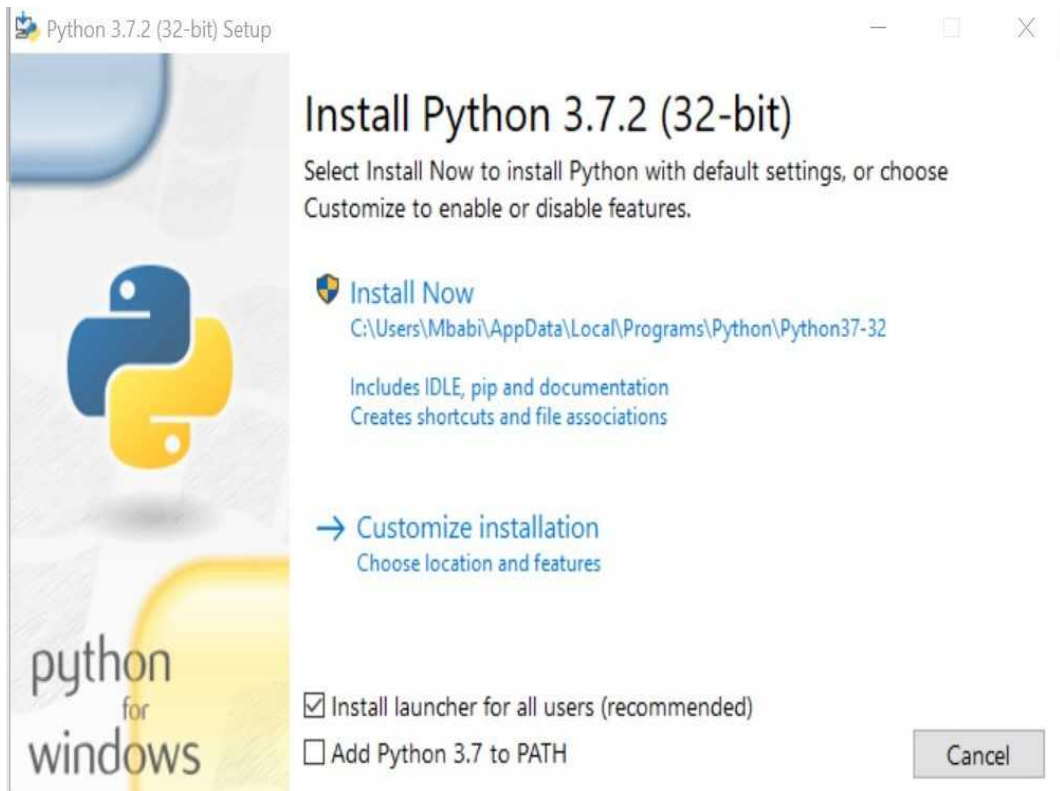
Additionally, if you are working offline, you must download the `check.py` module needed for testing on assignments. This is only necessary if you want to work offline. This file **must be in the same folder as the assignment problem you are working on**. This is the most common cause of the `No module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

Instructions for Windows Users

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

1	<p>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.</p>  <p>Download the latest version for Windows</p> <p>Download Python 3.7.2</p> <p>Looking for Python with a different OS? Python for Windows, Linux/UNIX, Mac OS X, Other</p> <p>Want to help test development versions of Python? Pre-releases</p> <p>Looking for Python 2.7? See below for specific releases</p>
2	<p>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.</p>

- 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 (recommmended)".

- 4 You will now be guided through the Wing 101 v7 installtion 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.

Download Wing 101:

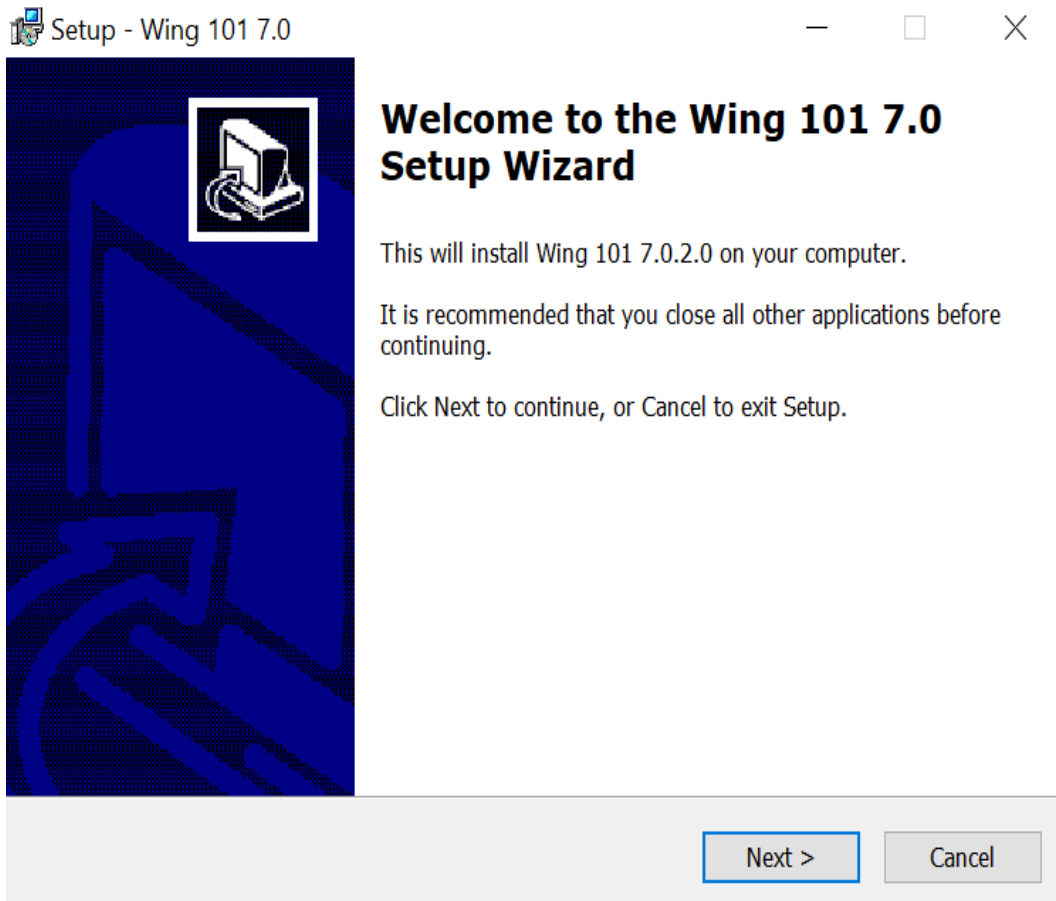
Windows Installer
32-bit and 64-bit

SHA1: 551bba903bb50e0b0bb270ea9fb9074cf6d3c752

Windows Zip File
32-bit and 64-bit

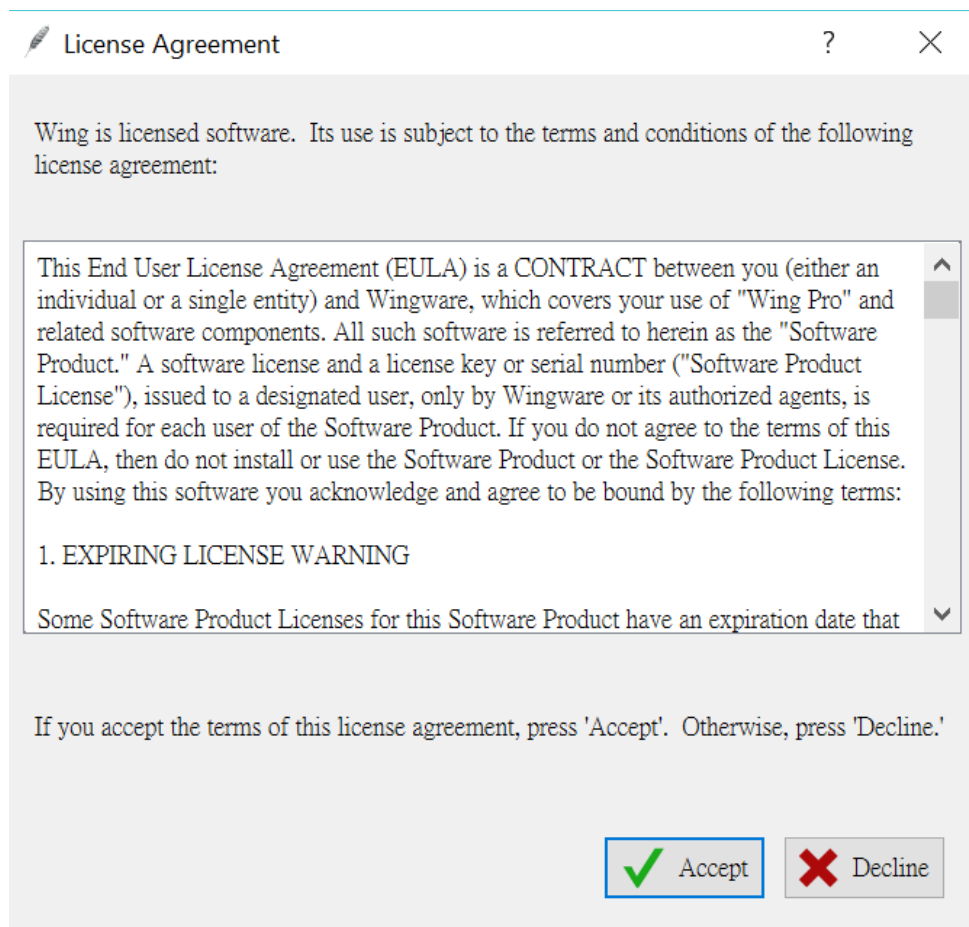
SHA1: 03468519f51b0164323aac35ea9ed391cc1e55af

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

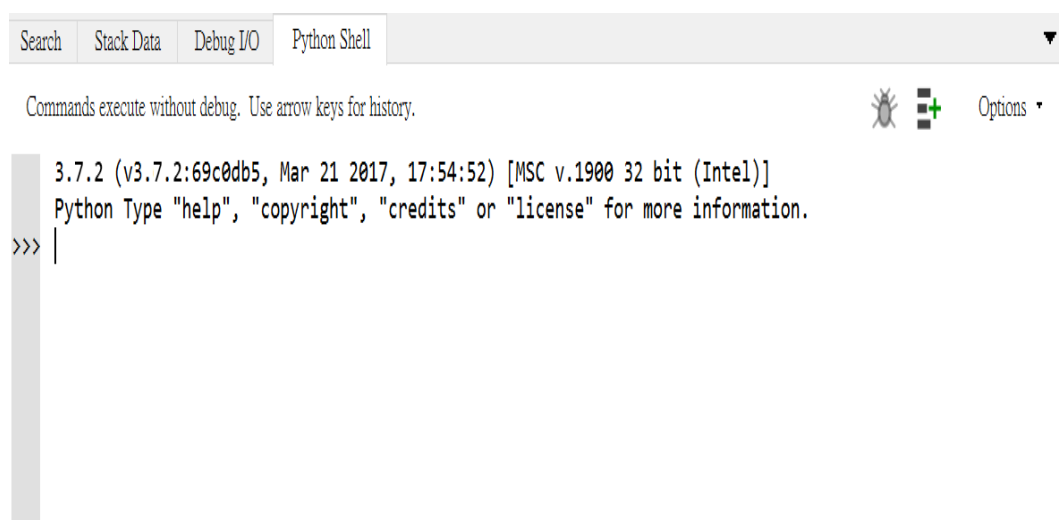


- 7 Follow the steps of the installer to install Wing 101. Make sure to agree to the license agreement to be able to use Wing.

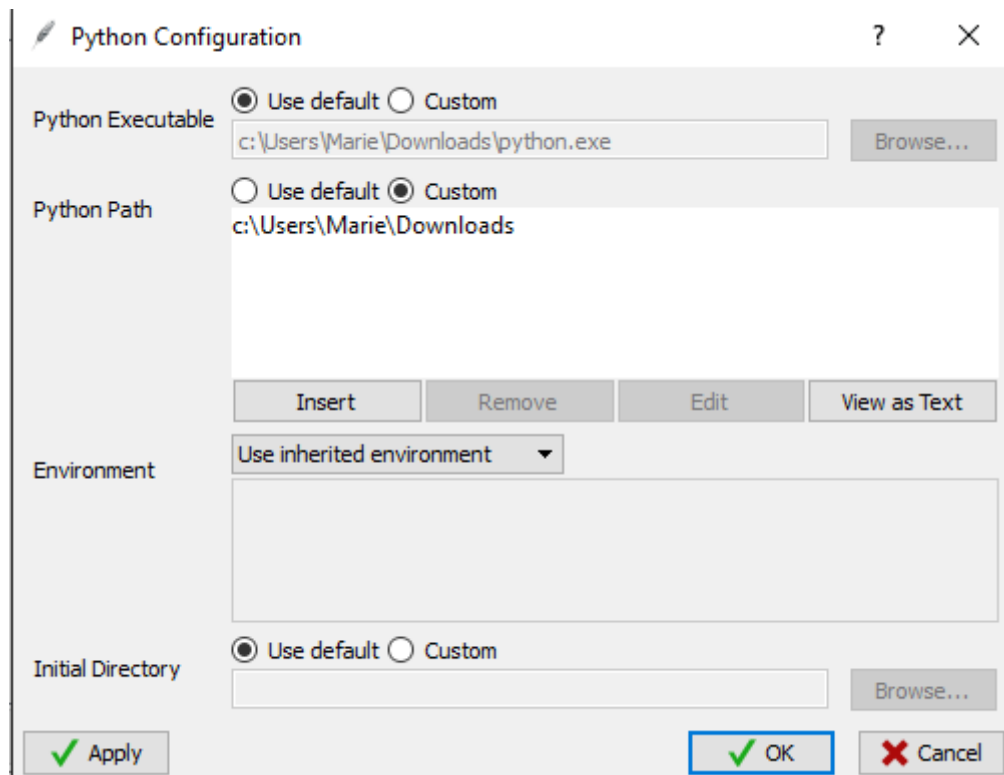
- 8 Once the installation is finished, you might need to lookup Wing 101 in your programs to be able to open it.
Open up Wing 101, and you should get the "License Agreement" window (first time only). Press "Accept" to enable access to the IDE.



- 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 corresponding configuration fields on the right. The 'Python Executable' section has radio buttons for 'Use default' (selected) and 'Custom', with a text field showing 'c:\Users\Marie\Downloads\python.exe' and a 'Browse...' button. The 'Python Path' section has radio buttons for 'Use default' and 'Custom' (selected), with a text field showing 'c:\Users\Marie\Downloads'. Below this are buttons for 'Insert', 'Remove', 'Edit', and 'View as Text'. The 'Environment' section has a dropdown menu set to 'Use inherited environment' and an empty text field below it. The 'Initial Directory' section has radio buttons for 'Use default' (selected) and 'Custom', with an empty text field and a 'Browse...' button. At the bottom are three buttons: 'Apply' with a green checkmark, 'OK' with a green checkmark and a blue border, and 'Cancel' with a red X.

Python Configuration ? X

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

Python Path ☐ Use default ☒ Custom
c:\Users\Marie\Downloads
Insert Remove Edit View as Text

Environment Use inherited environment ▼

Initial Directory ☒ Use default ☐ Custom
Browse...

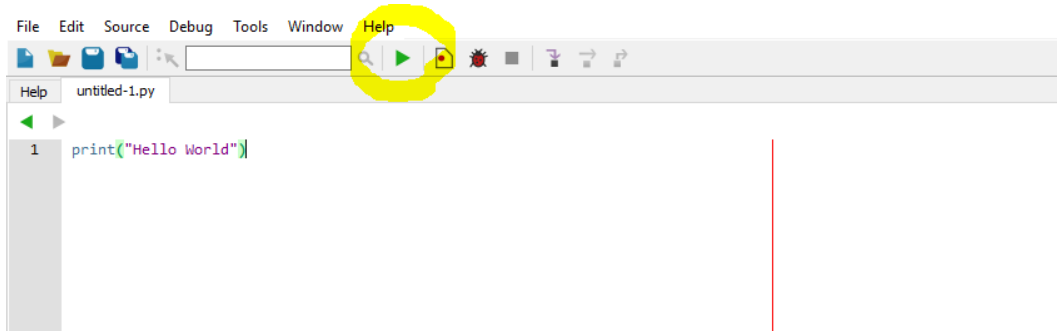
✓ Apply ✓ OK ✗ Cancel

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

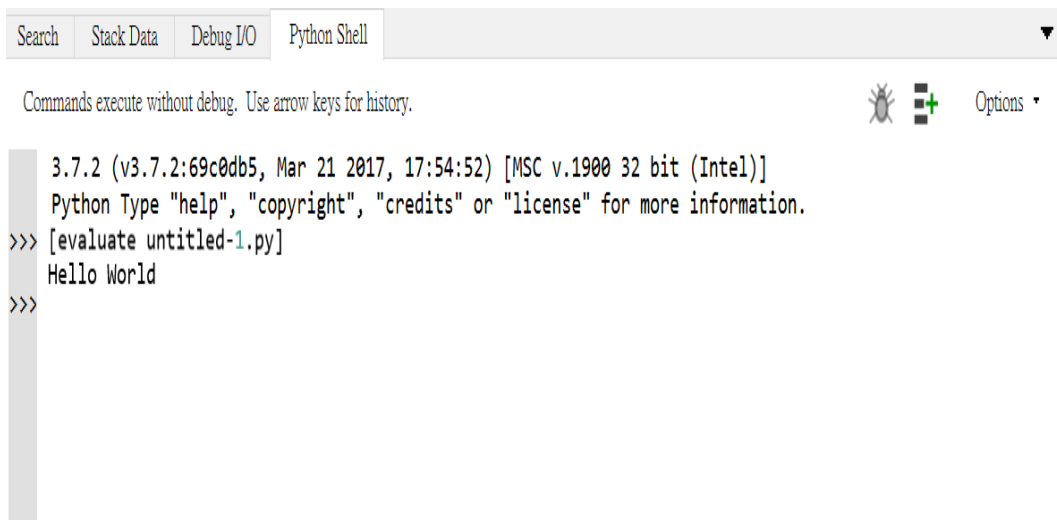
The image displays two screenshots of the 'Python Configuration' dialog box. The top screenshot shows the 'Python Executable' field with the path 'c:\Users\Marie\Downloads\python.exe' and the 'Custom' radio button selected. The 'Browse...' button is visible next to the field. The bottom screenshot shows the same dialog box, but the 'Browse...' button is highlighted with a yellow circle, indicating the next step in the process.

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.

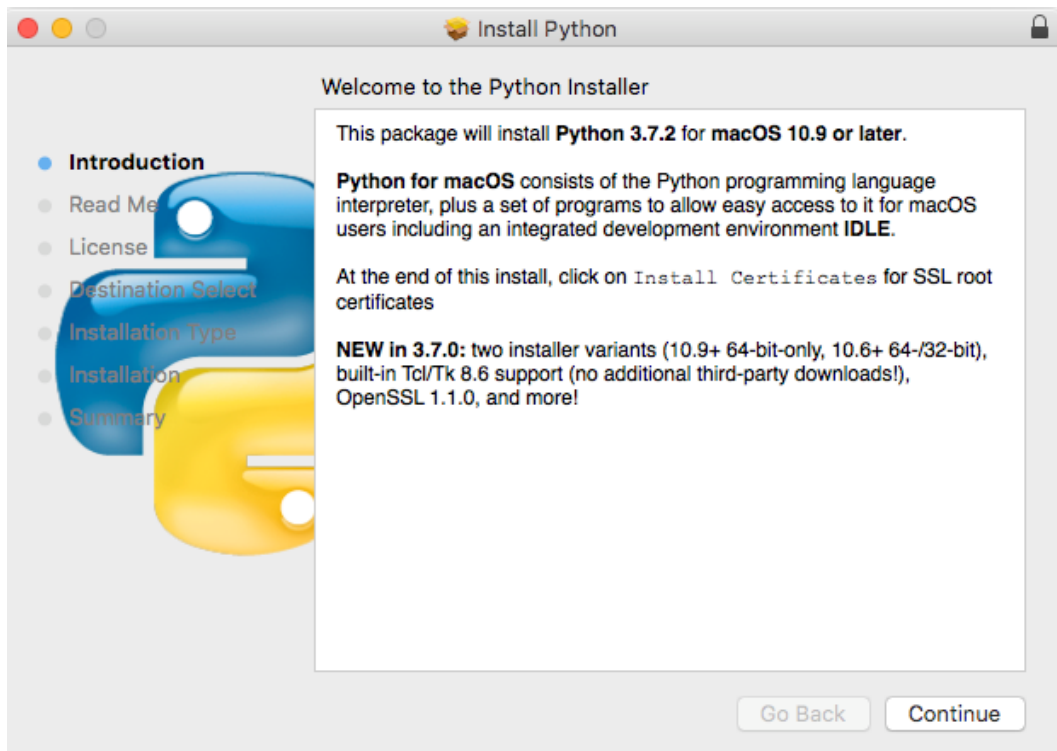


- 1 Go to the Python Download page, which is located at <https://www.python.org/downloads>. You should see the following on the top of the page. Python 3 still needs to be installed, even though Python 2 should already be installed.



- 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

SHA1: b6eda346b25f7c8c22d8a073e3ece277f8d99519

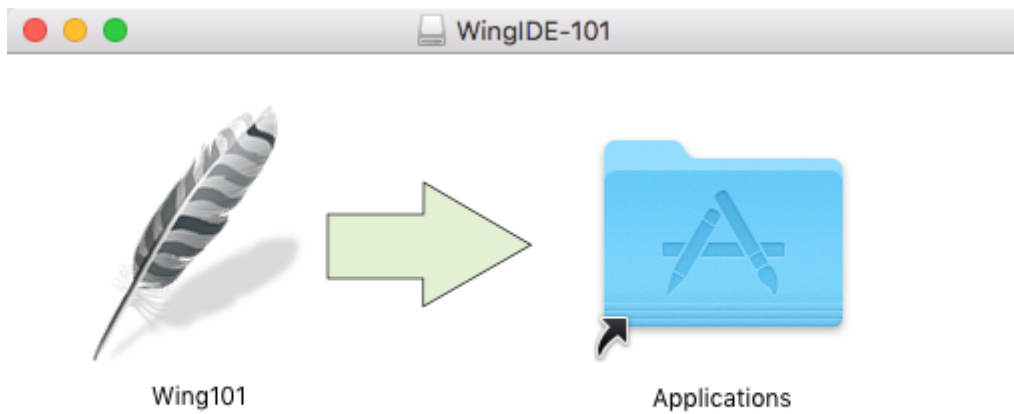
[Supported OSes](#)
[Supported Python Versions](#)

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

Download Wing 101:

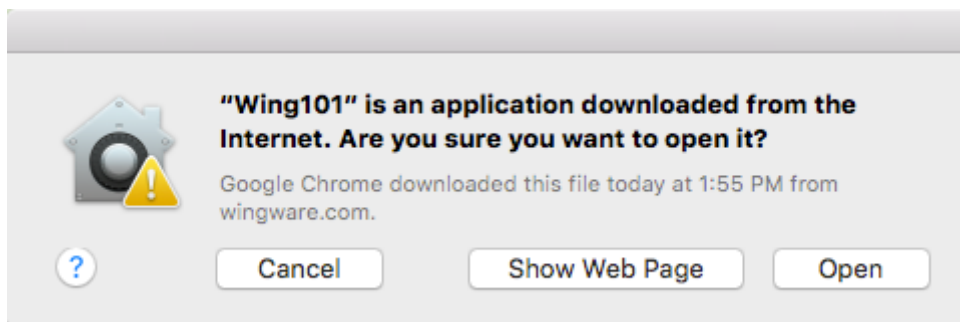
OS X Installer
64-bit Intel

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



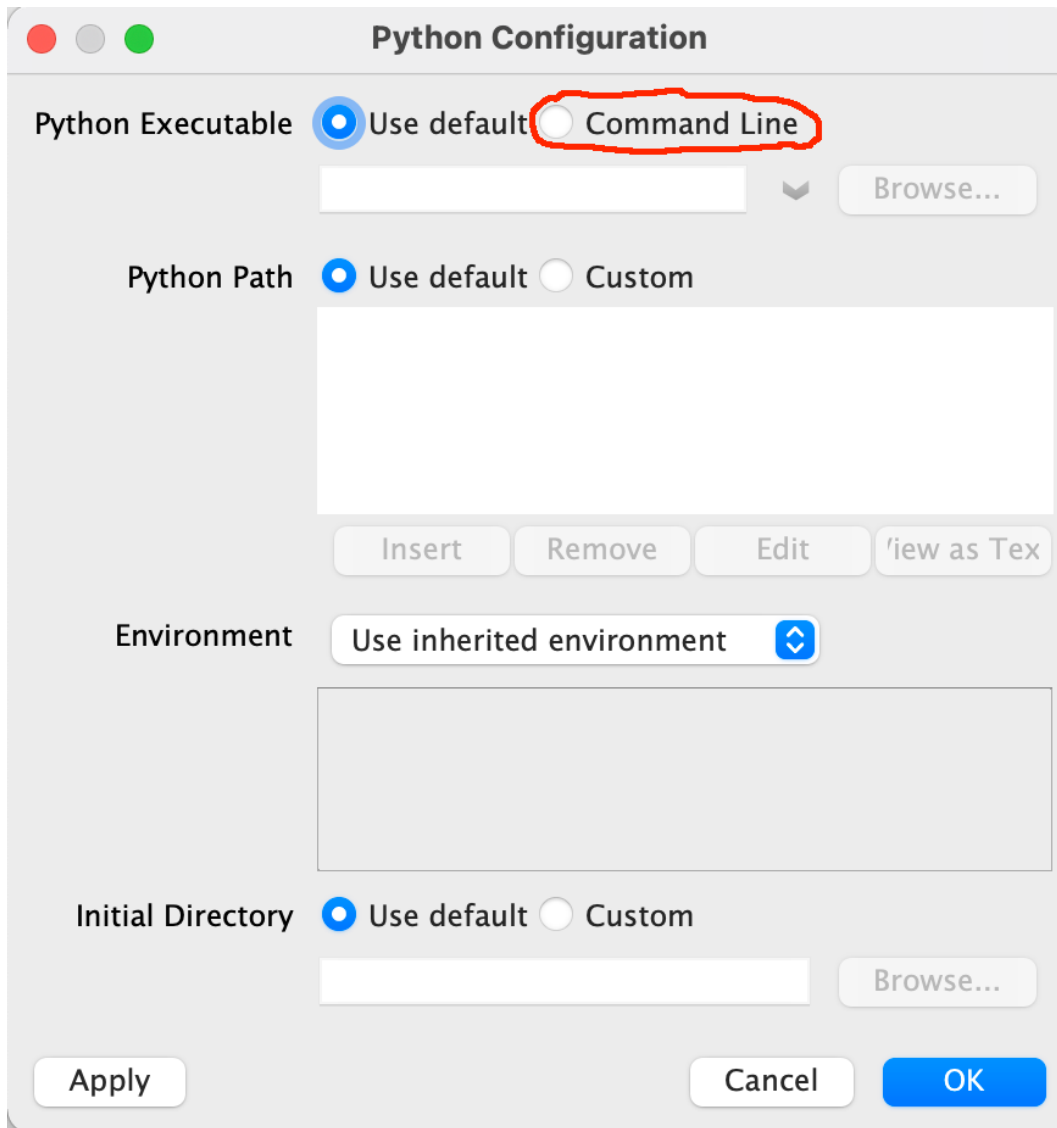
Follow the arrow to drag Wing101 to the **Applications** folder. It might take a few minutes to finish copying.

- 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:

The image shows a 'Python Configuration' dialog box with a title bar containing three colored buttons (red, grey, green). The dialog is divided into several sections. The first section, 'Python Executable', has two radio buttons: 'Use default' (selected) and 'Command Line' (circled in red). Below these is a text field and a 'Browse...' button. The second section, 'Python Path', has two radio buttons: 'Use default' (selected) and 'Custom'. Below these is a large text area and three buttons: 'Insert', 'Remove', and 'Edit'. The third section, 'Environment', has a dropdown menu set to 'Use inherited environment' and a large empty text area. The fourth section, 'Initial Directory', has two radio buttons: 'Use default' (selected) and 'Custom'. Below these is a text field and a 'Browse...' button. At the bottom of the dialog are three buttons: 'Apply', 'Cancel', and 'OK'.

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.)

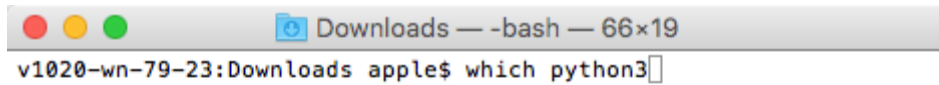
- 9 You will now be guided to find the location of the Python Executable.
Open up **Terminal** on your mac.



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

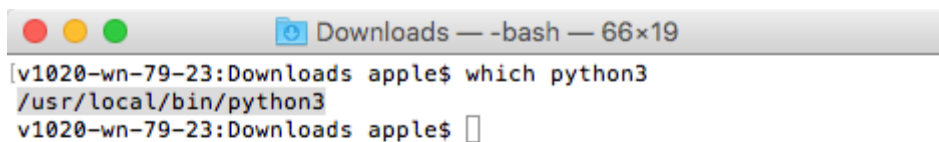
- (1) The location of **Terminal** is usually: **Applications/Utilities/Terminal**. You may find the exact location and open it.
- (2) Using **Spotlight Search** (the magnifying glass icon on the top-right corner of the menu bar) to search for **Terminal** on your Mac and then open it.

- 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:

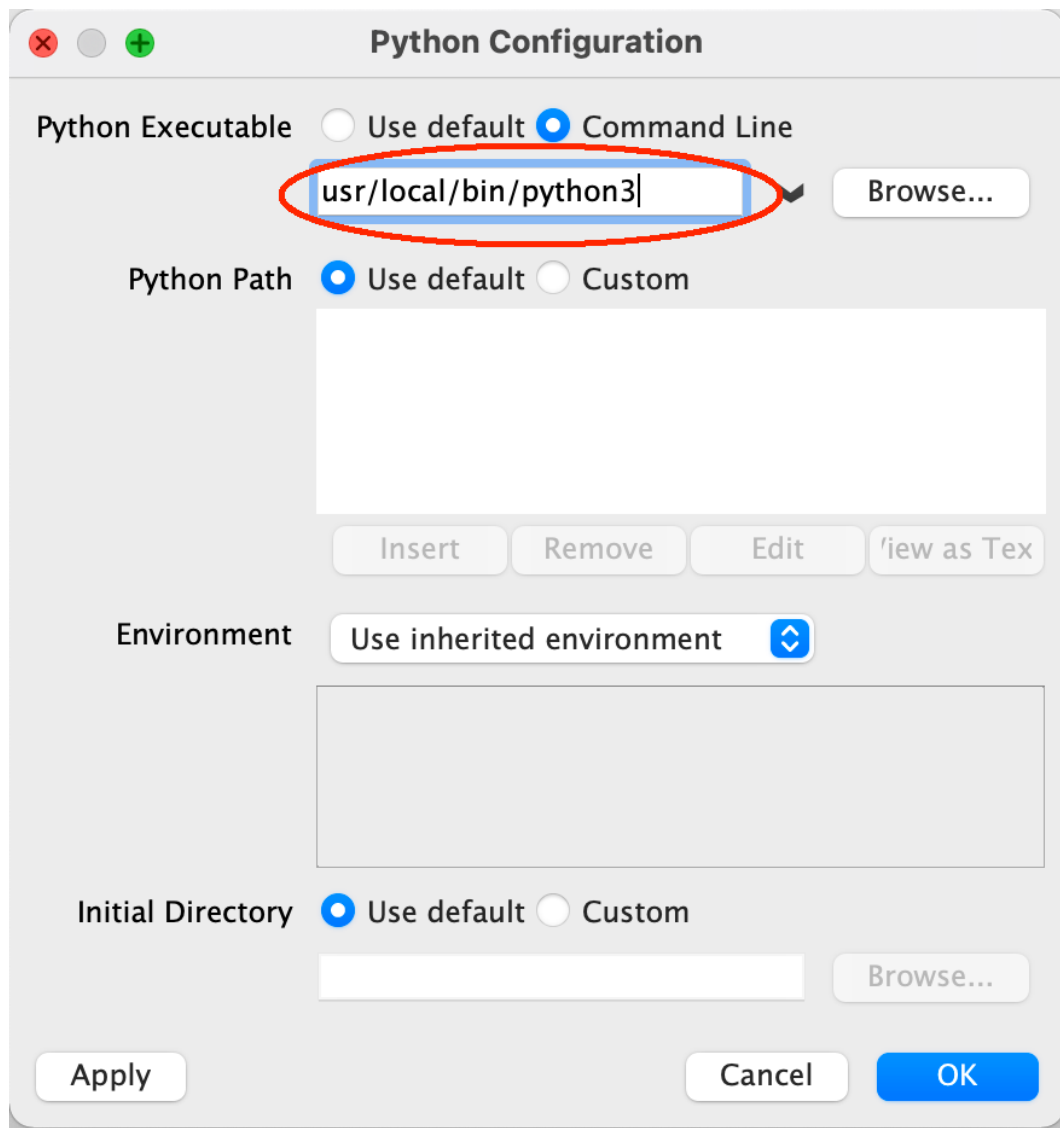


```
v1020-wn-79-23:Downloads apple$ which python3
/usr/local/bin/python3
v1020-wn-79-23:Downloads apple$
```

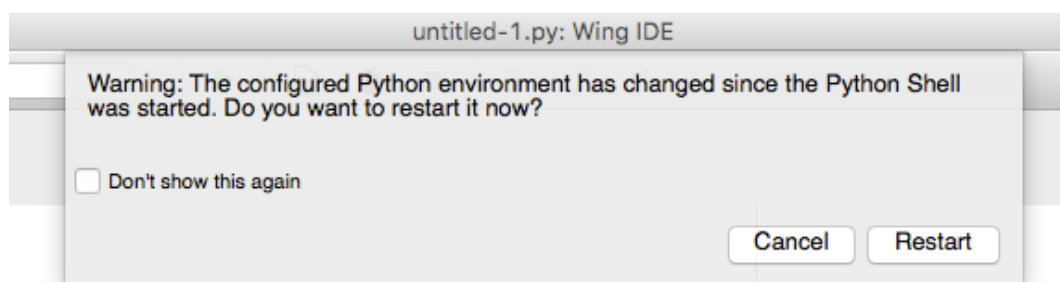
Copy the highlighted line.

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

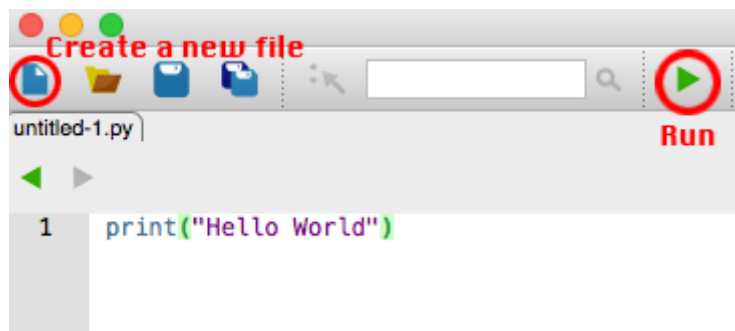
- 11 Go back to Wing101 and **paste** the address under Custom option, as circled in red in the following image. Then, click **Apply**.



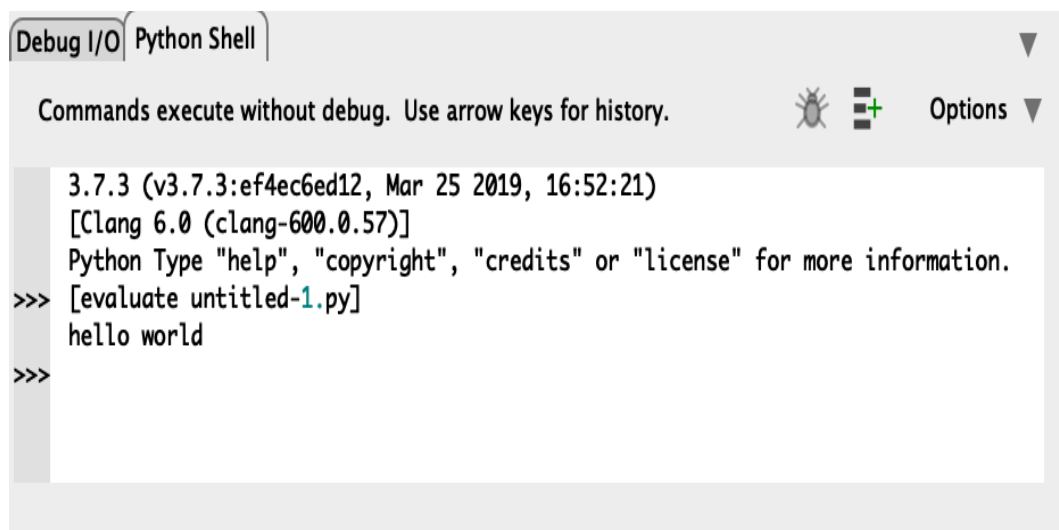
The following warning will show up, then click **Restart**.



- 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

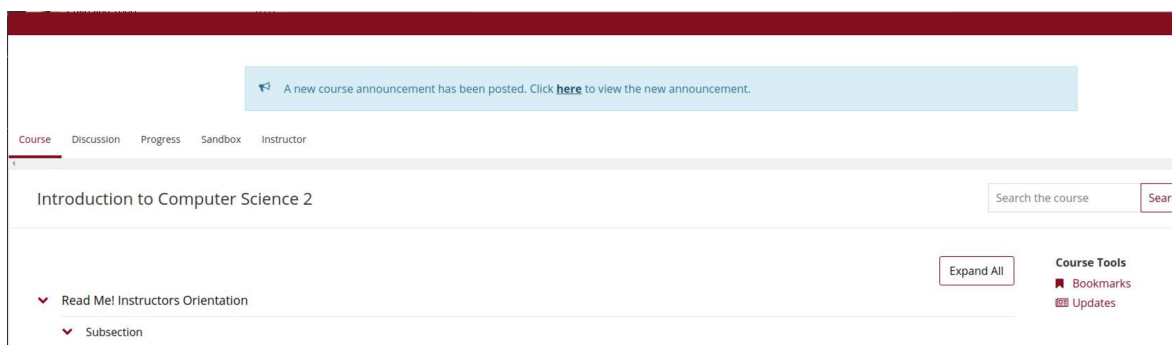
If you are experiencing any technical difficulties with viewing content on EdX (for example, videos not loading etc.) please contact cs-online@uwaterloo.ca who can help you debug. Please do not contact this address with course content related issues (post on our forums or ask instructors - see below)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

Carmen Bruni. Carmen is a lecturer at the University of Waterloo and has been since 2015. He started with the Centre for Education in Mathematics and Computing and moved to the David R. Cheriton School of Computer Science in 2017. He has a Ph.D. in Mathematics from the University of British Columbia studying extension of Fermat's Last Theorem. Carmen has been teaching



CS116 since 2017 and is the primary course author for CS116. In his spare time, Carmen enjoys playing board games and watching football, chess and speed running video games. Email: cbruni@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. 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 or the preferred method via the EdX discussion forums. Below are the ISAs for this course:



Dhruv Babu. Dhruv is a 3A Math student following the Mathematical Physics major plan. His previous co-op experience includes working as a Senior Online Learning Assistant for the Faculty of Mathematics. Dhruv has an interest in rocketry and astronomy and has been part of the Waterloo Rocketry Design Team since 2020. Some hobbies Dhruv enjoys include playing guitar, reading science fiction novels, and practicing archery.



Wendy Li. Wendy is a second year Math student in Mathematical Finance major. Wendy has many hobbies including badminton, archery, drawing, and playing Pipa (a traditional Chinese instrument).

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:



Daniel Erhabor. Daniel is in the Master of Mathematics in Computer Science program and has been since January 2021. His current research is on how systems programmers use GitHub CoPilot. He has a Bachelor of Science in Computer Science from Howard University. He is a cultured anime watcher and believes "Overlord" and "That Time I was Reincarnated as a Slime" are two of the best Isekai animes

of ALL time.



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 enabling application-aware resource management with accelerators in serverless. He is engaged in entrepreneurship, building trading algorithms, and prototyping a personal finance mobile application. In his free time, Justin enjoys playing the piano and improving his skills in disc golf and soccer.

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	Content Due Date	Assignment Due Date
1	Wednesday September 7th, 2022 at 12:01am ET	Tuesday September 13th, 2022 at 10:00am ET	Thursday September 15th, 2022 at 10:00am ET
2	Monday September 12th, 2022 at 12:01am ET	Tuesday September 20th, 2022 at 10:00am ET	Thursday September 22nd, 2022 at 10:00am ET
3	Monday September 19th, 2022 at 12:01am ET	Tuesday September 27th, 2022 at 10:00am ET	Thursday September 29th, 2022 at 10:00am ET
4	Monday September 26th, 2022 at 12:01am ET	Tuesday October 18th, 2022 at 10:00am ET	Thursday October 20th, 2022 at 10:00am ET
5	Monday October 3rd, 2022 at 12:01am ET	Tuesday October 25th, 2022 at 10:00am ET	Thursday October 27th, 2022 at 10:00am ET
6	Monday October 17th, 2022 at 12:01am ET	Tuesday November 1st, 2022 at 10:00am ET	Thursday November 3rd, 2022 at 10:00am ET
7	Monday October 24th, 2022 at 12:01am ET	Tuesday November 8th, 2022 at 10:00am ET	Thursday November 10th, 2022 at 10:00am ET
8	Monday October 31st, 2022 at 12:01am ET	Tuesday November 15th, 2022 at 10:00am ET	Thursday November 17th, 2022 at 10:00am ET
9	Monday November 7th, 2022 at 12:01am ET	Tuesday November 22nd, 2022 at 10:00am ET	Thursday November 24th, 2022 at 10:00am ET
10	Monday November 14th, 2022 at 12:01am ET	Tuesday November 29th, 2022 at 10:00am ET	Thursday December 1st, 2022 at 10:00am ET

Note that the Content Due date includes all of the Warm-Up, Wrap-Up and Concept Check questions in the module.

Detailed Daily Course Schedule

In this section we give our recommendation for what to work on every day of the term (weekdays,

excluding holidays and reading week). You are not required to adhere rigidly to this schedule; it's offered based on our expectation for how long it will take to complete the individual components of the course. We think setting aside roughly 2 hours a day for this course should make the following schedule feasible.

Detailed Daily Course Schedule

In this section we give our recommendation for what to work on every day of the term (weekdays, excluding holidays and reading week). You are not required to adhere rigidly to this schedule; it's offered based on our expectation for how long it will take to complete the individual components of the course. We think setting aside roughly 2 hours a day for this course should make the following schedule feasible.

Date	Recommended Course content	Deadlines
W: September 7	Orient Yourself with the Course, Style Guide, Syllabus Sections, M1 Warm-Up Quiz, Read Assignment 1	
R: September 8	Lesson 1.1, Lesson 1.2, Lesson 1.3	
F: September 9	Lesson 1.4, Lesson 1.5, Start Assignment 1	
M: September 12	M1 Wrap-Up Quiz, Continue Assignment 1	
T: September 13	Continue Assignment 1	Module 1 Quizzes and Concept Checks Due at 10:00am ET
W: September 14	M2 Warm-Up Quiz, Lesson 2.1, Read Assignment 2	
R: September 15	Lesson 2.2, Lesson 2.3	Assignment 1 Due at 10:00am ET
F: September 16	Lesson 2.4, M2 Wrap-Up Quiz, Start Assignment 2	
M: September 19	Continue Assignment 2	
T: September 20	Continue Assignment 2	Module 2 Quizzes and Concept Checks Due at 10:00am ET
W: September 21	M3 Warm-Up Quiz, Lesson 3.1	
R: September 22	Lesson 3.2, Lesson 3.3	Assignment 2 Due at 10:00am ET
F: September 23	Lesson 3.4, Start Assignment 3	

M: September 26	Lesson 3.5 and Wrap-Up Quiz, Continue Assignment 3	
T: September 27	Continue Assignment 3	Module 3 Quizzes and Concept Checks Due at 10:00am ET
W: September 28	Continue Assignment 3	
R: September 29	M4 Warm-Up Quiz, Lesson 4.1, Read Assignment 4	Assignment 3 Due at 10:00am ET
F: September 30	Lesson 4.2 and 4.3	
M: October 3	Lesson 4.4 and 4.5	
T: October 4	Lesson 4.6 and Lesson 4.7	
W: October 5	M4 Wrap-Up Quiz, Start Assignment 4	
R: October 6	Continue Assignment 4	
F: October 7	M5 Warm-Up Quiz, Lesson 5.1, Read Assignment 5	
October 10-14	Reading Week	
M: October 17	Lesson 5.2	
T: October 18	Lesson 5.3	Module 4 Quizzes and Concept Checks Due at 10:00am ET
W: October 19	M5 Wrap-Up Quiz, Start Assignment 5	
R: October 20	Continue Assignment 5	Assignment 4 Due at 10:00am ET
F: October 21	M6 Warm-Up Quiz, Read Assignment 6	
M: October 24	Lesson 6.1	
T: October 25	Lesson 6.2	Module 5 Quizzes and Concept Checks Due at 10:00am ET
W: October 26	Lesson 6.3, M6 Wrap-Up Quiz	

R: October 27	Start Assignment 6	Assignment 5 Due at 10:00am ET
F: October 28	Continue Assignment 6	
M: October 31	M7 Warm-Up Quiz, Lesson 7.1, Read Assignment 7	
T: November 1	Lesson 7.2 and 7.3	Module 6 Quizzes and Concept Checks Due at 10:00am ET
W: November 2	Lesson 7.4 and 7.5	
R: November 3	M7 Wrap-Up Quiz, Start Assignment 7	Assignment 6 Due at 10:00am ET
F: November 4	Continue Assignment 7	
M: November 7	M8 Warm-Up Quiz, Read Assignment 8	
T: November 8	Lesson 8.1, Lesson 8.2	Module 7 Quizzes and Concept Checks Due at 10:00am ET
W: November 9	Lesson 8.3, Lesson 8.4	
R: November 10	Lesson 8.5, Lesson 8.6	Assignment 7 Due at 10:00am ET
F: November 11	M8 Wrap-Up Quiz, Start Assignment 8	
M: November 14	Continue Assignment 8	
T: November 15	M9 Warm-Up Quiz, Read Assignment 9	Module 8 Quizzes and Concept Checks Due at 10:00am ET
W: November 16	Lesson 9.1	
R: November 17	Lesson 9.2	Assignment 8 Due at 10:00am ET
F: November 18	Lesson 9.3	
M: November 21	M9 Warm-Up Quiz, Start Assignment 9	

T: November 22	Continue Assignment 9	Module 9 Quizzes and Concept Checks Due at 10:00am ET
W: November 23	M10 Warm-Up Quiz, Lesson 10.1, Read Assignment 10	
R: November 24	Lesson 10.2, Lesson 10.3	Assignment 9 Due at 10:00am ET
F: November 25	M10 Wrap-Up Quiz, Start Assignment 10	
M: November 28	Continue Assignment 10	
T: November 29	Continue Assignment 10	Module 10 Quizzes and Concept Checks Due at 11:59pm ET
W: November 30	Continue Assignment 10	
R: December 1	Read Final Exam Module and do practice problems	Assignment 10 Due at 10:00am ET
F: December 2	Read Final Exam Module and do practice problems	
December 5th Week	Prepare For Final Exam	Check Announcements for Final Exam Date

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

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 Quizzes	5%
Concept Checks	10%
Wrap-Up Quizzes	5%
Assignments	30%
Final Exam	50%

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

More information about setting up a proctor through CEL can be found here <https://uwaterloo.ca/extended-learning/learn-online/write-exams>.

In the event, and only in the event, that an in person final exam **cannot** be written for everyone (say due to a global pandemic), we will adjust the above grading scheme to the following grading scheme:

Component	Weight (%)
Warm-Up Quizzes	10%
Concept Checks	20%
Wrap-Up Quizzes	10%
Assignments	60%

Important! The grading schemes are **not interchangeable** in the sense that you **do not** get to pick which scheme to use. If an in person final exam can be written then the scheme including the final exam will be the scheme used for the course. If an individual is unable to write the final exam, proper documentation and consultation with the ISC can result in an INC and the exam can be written at a future time (say in a future term).

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.

Concept Checks

Your concept check grade is computed as the sum total points earned over all problems completed divided by 90% of the total number of points for all concept checks in the entire course up to a maximum of 100%. Some modules have more questions than others and hence have more questions so to not overemphasize simpler modules like module 1, we want to weigh the concept checks questions equally throughout the course. Questions are also potentially worth different amounts so it is worth checking how much a problem is worth.

The Concept Check questions will each have two attempts for non-programming problems and unlimited attempts for programming problems. This allows you to make mistakes while learning the content and still obtain a good grade. Some of these are intentional — questions that are a bit beyond what you can do in order to reinforce understanding and some are questions that have common misconceptions that are meant to trick you in order to resolve a misconception. Again, do not be worried about getting perfect on everything because you are not expected to. Mistakes are going to happen because you are learning and multiple attempts for these concept check questions is meant to allow for some of these mistakes to occur. If you follow through all of the coursework, start assignments early, and ask for help, you should be able to do really well in this course.

Please note that Concept Check questions **will occur in required videos** and so you should also watch the required videos to obtain full marks on these problems. Videos must be watched from within the browser in order to activate the questions.

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.

There will be no extensions on assignments. If sick, please email our ISC a sick note to discuss alternate arrangements. Reweighting of assignments is not automatic even with a valid doctor's note and is up to the sole discretion of the ISC or instructor to allow for reweighting.

Assignments are weighted evenly.

We will also only take your best 8 assignments of the first 9 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.

Assignment 10 cannot be dropped and will count towards your final grade. Thus your assignment grade will be computed as your best first 8 of 9 assignments and assignment 10.

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 a sick note to discuss alternate arrangements. Reweighting of assignments is not automatic even with a valid doctor's note and is up to the sole discretion of the ISC or instructor to allow for reweighting. We will only accept doctor's notes on at most 3 assignment components so make sure you are on top of your studies.

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 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

- 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/ 519-888-4567 ext 32655
- MATES: one-to-one peer support program offered by Federation of Students (FEDS) and Counselling Services: 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.

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.

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.

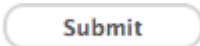
1. On your MarkUs homepage, click the assignment name to go to its submission page.

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 indicted 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

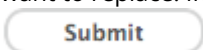
2. Click the "Submissions" tab at the top of the page.
3. Click "Add A New File". A new row will be added to the table. Click "Choose File" or "Submit" in the new row. (The button's name will depend on your web browser.) In the window that appears, browse to where you have saved your file and select 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.

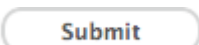
Replacing a file

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

All office hours will be done using MS Teams. Instructions for this can be found below the office hour timetable.

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 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.

Please note - Occasionally on Mondays at 12:30 we will have a brief team meeting so responses to student inquiries might be a bit slower at this time. Thank you for your understanding.

Timetable

	Monday	Tuesday	Wednesday	Thursday	Friday
8:00 AM					
8:30 AM					
9:00 AM					
9:30 AM					
10:00 AM	Carmen Bruni	Carmen Bruni			
10:30 AM	10:00 - 11:00	10:00 - 11:00			
11:00 AM	Dhruv Babu	Wendy Li			
11:30 AM	11:00 - 12:00	11:00 - 12:00			
12:00 PM	Carmen Bruni	Carmen Bruni		Justin San Juan	Dhruv Babu
12:30 PM					

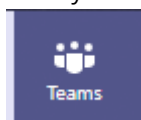
	12:00 - 2:00	12:00 - 2:00		12:00 - 2:00	12:00 - 2:00
1:00 PM			Justin San Juan		
1:30 PM			1:00 - 3:00		
2:00 PM		Wendy Li		Wendy Li	Justin San Juan
2:30 PM		2:00 - 4:00		2:00 - 4:00	2:00 - 3:00
3:00 PM			Dhruv Babu		
3:30 PM			3:00 - 5:00		
4:00 PM		Daniel Erhabor		Daniel Erhabor	
4:30 PM		4:00 - 6:00		4:00 - 6:00	
5:00 PM			Daniel Erhabor		
5:30 PM			5:00 - 6:00		
6:00 PM					
6:30 PM					
7:00 PM					
7:30 PM	Carmen Bruni				
8:00 PM	7:30 - 9:30				
8:30 PM					
9:00 PM					

Instructions

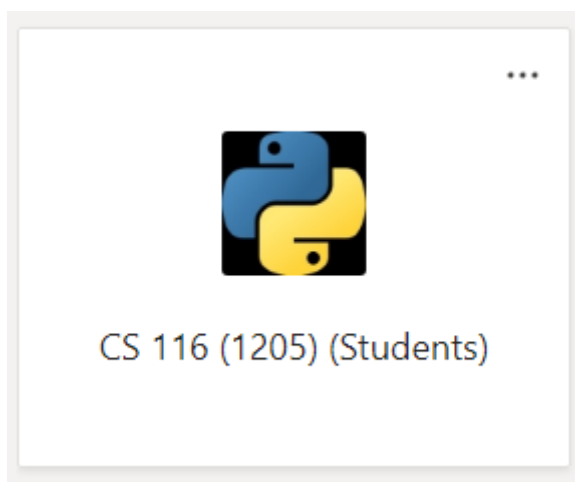
You can use Teams through your browser at teams.microsoft.com or you can download the app from [Microsoft's download page](#).

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 (1211) (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.

< All teams



CS 116 (1209) (Students)

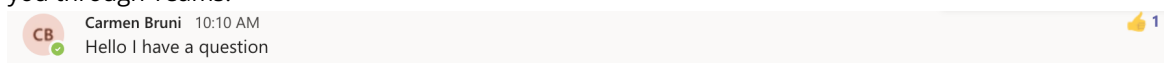


General



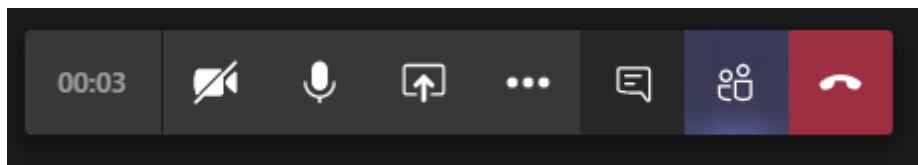
We will have one general chat channel where everyone will congregate to get their questions answered.

At the bottom, there is a section labeled "Start a new conversation". That's where you'll be posting to let us know you'd like to talk. As soon as we're able, we will give you a thumbs up as shown below and call you through Teams.



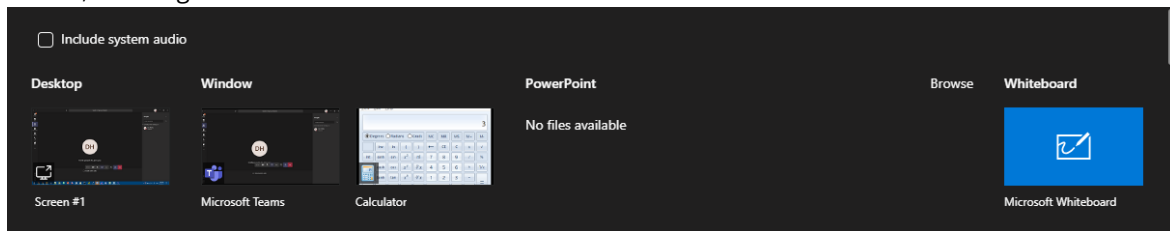
The Thumbs up help both instructors and students keep track of where we are in the office hour queue. It is a video call, but you do not need to use a webcam. We are using video calls so that you can show us your screen. If you do not have a microphone, you can use the text chat during the call.

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.

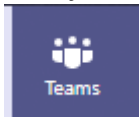
If office hours are busy, we may impose a time limit on your call (you can always go to the back of the line).

Instructions

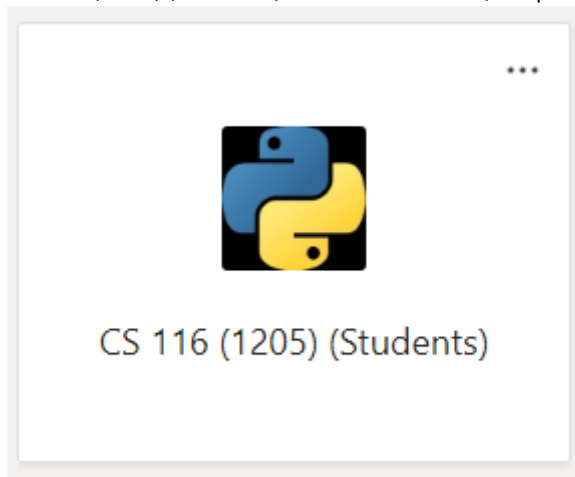
You can use Teams through your browser at teams.microsoft.com or you can download the app from [Microsoft's download page](#).

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 (1211) (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.

< All teams



CS 116 (1209) (Students)

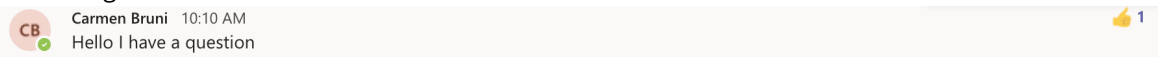


General



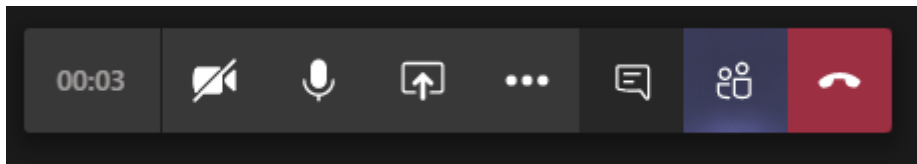
We will have one general chat channel where everyone will congregate to get their questions answered.

At the bottom, there is a section labeled "Start a new conversation". That's where you'll be posting to let us know you'd like to talk. As soon as we're able, we will give you a thumbs up as shown below and call you through Teams.



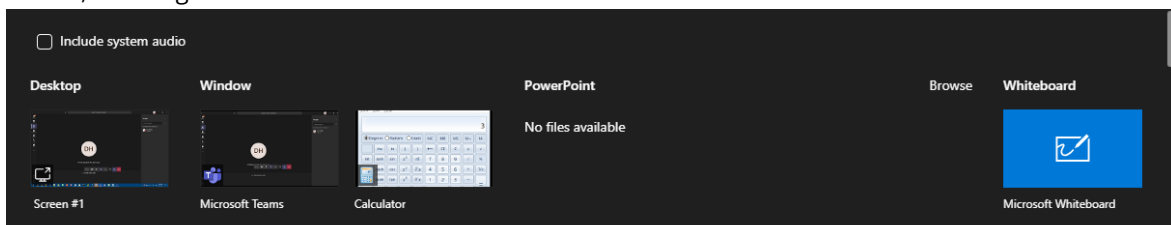
The Thumbs up help both instructors and students keep track of where we are in the office hour queue. It is a video call, but you do not need to use a webcam. We are using video calls so that you can show us your screen. If you do not have a microphone, you can use the text chat during the call.

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).