Syllabus: Introduction to Computer Science
COSC 1000

Instructor:	Thomas Schwarz, SJ (thomas.schwarz@marquette.edu)
Mode:	Presence / Synchronous Long Distance
Contents:	This course is under re-development. Half the course will convey the breadth and current status of computer science, the other half will provide basic programming skills in Python and with Arduino. The latter aims to provide an understanding of programming more than a professionally exploitable skill, though it is of course hoped that interested students will find this a sufficient basis to develop these skills either at Marquette University or through other means. The first half will pay particular attention to the interaction between computing and culture and economy. As the course is under development, student feedback is encouraged and the topics are fluid.
Long Distance Mode:	Long distance learning is synchronous. You will use the provided Zoom link to access the class in real time. Please consider enabling video, but disable audio unless you want to ask a question. While Marquette has specialized class rooms, be advised that the setup is not ideal. Past experience has shown that Covid infection even among the young can be quite debilitating. For this reason, classes will be taped and the contents made available. Be mindful of this if you ask questions. Video information will not be shared, unless you ask explicitly for screen sharing.
Web site:	Because of my set-up, I am using http://tschwarz.mscs.mu.edu (look for classes) as my principal way of content dissemination.
Academic Procedures:	All Marquette policies apply even without explicit mention in the syllabus. In particular, the equivalence of two weeks of absence from the class can lead to an automatic drop. Academic misconduct can lead to sanctions. In case of doubt, it is always preferable to contact me. Example: You work with another student on a project, even though it is not explicitly allowed. This can be interpreted as academic misconduct. Much easier, talk to me. Even if you do so after the fact, but before it comes to my attention, problems can be avoided.

Covid and similar illnesses:	I will try to accommodate any absences caused by Covid (or similar illnesses), but I am restricted by limitations on my time and by Marquette's very rigorous time tables. It is your duty as a student to inform me of exceptional circumstances such as the inability to work for more than a few days as soon as possible.
Computing resources	I expect you to have access to a computer that (a) can install Vanilla Python, (b) has a USB drive, and (c) can install the Arduino programming environment. Pretty much any computer (WinOS, MacOS, Linux) will do, unless you have a Chrome book with very limited storage. **If you cannot afford a computer with these requirements, talk to me as soon as possible.** Also, if you have a usable computer you do not need, talk to me about a donation to the Jesuit loaner program.
Books:	In lieu of a text book, you will need to have an Arduino and a few electronic components such as LEDs, resistors, and a small breadboard. I have good experiences with the starter kits by Elegoo (with an Arduino clone). You will be able to share the kit with a colleague in class.
Grading:	Based on tri-weekly quizzes (administered via D2L) 20%, a midterm and a final (20%+20%), an Arduino project (10%) and a collection of small Python projects (30%). If you can do so safely, you can do the Arduino project in a group of two or three with more expected content. The small Python projects are individual work only. There is a "redo" policy for Python projects.

Planned content:

Week 1: Early history of computing, Installing Python and using IDLE for calculations
Week 2: Computer Organization 1. Writing simple Python scripts
Week 3: Computer Organization 2. Python Input and output.
Week 4: Storage Systems. Network Effect. Python program flow control 1.
Week 5: Operating Systems. Python program flow control 2.
Week 6: Networking 1. Python functions 1.
Week 7: Networking 2. Python functions 2.
Week 8: Numeric computations. Lists in Python.
Week 9: Database systems. Tuples in Python.
Week 10: Machine learning. Strings in Python.
Week 11: Machine learning. String processing in Python.
Week 12: Artificial Intelligence. Arduino introduction.
Week 13: Computer Forensics. Arduino project.
Week 14: Computer Security. File Systems in Python.
Week 15: Distributed Systems. Numpy and Pandas.