CPSC110H: Honors Introduction to Computer Science

essentials

#	resource
1	[The current version of this syllabus](#)
2	Instructor: Ron Zacharski ron.zacharski@gmail.com, 575.680.4041
3	[Inquiryum arduino course](#)
4	[Experience Point Sheet](#)
5	The [Arduino Programming Fundamentals Discord Server](#)

Here is what you should accomplish in the first week

1. Read this syllabus
2. Complete the [course signup form](#)
3. Join the [Arduino Programming Fundamentals Discord Server](#) and write a "hi" post.
4. Watch the initial videos on [the Inquiryum Arduino website.](#)
5. Get an LED light to blink following the course videos.

Description

This course provides an introduction to computer programming focusing on how to build physical devices and design programs.

Course Objectives

These objectives are true of **all sections** of cs110 – Intro to CS.

- Gain a conceptual understanding of computers, computing, and issues with technology through breadth-first topics.
- Develop skills in algorithmic problem-solving, expressed in a programming language.
- Represent and manipulate information in a program that executes on a computer.

Honor Program Learning Outcome

This course focuses on self-directed learning where you will be an active agent in your own education. The class is discovery-based and student-centered. The majority of class time will be spent working on projects. You will also be expected to do regular work outside of class. This class will help you make the transition from classroom learning to independent learning. In the other sections of this course, information is provided to you in long traditional lectures. In this section the majority of class time is open lab time. **This student-centered focus is why this course is designated honors.**

Honors section goal

The goal of this section is to acquire programming skills. In addition to the course objectives listed above, objectives specific to these sections include:

- to be able to use basic algorithm concepts to solve problems including
    - variables
    - boolean expressions
    - conditions
    - loops
    - events
    - functions
- to be able to write simple programs to solve problems

Format - an asynchronous online class

This class is asynchronous meaning that you will be working through the [Inquiryum Arduino Programming Fundamentals Course](). You can watch the videos anytime you want. You can play them at a faster speed, you can rewatch them or pause them. You can work on the course material in 20 minute blocks throughout a day, or devote a large contiguous block of time once per week. The advantages of this approach is that it allows you great flexibility in when you want to work on the material and for how long. And, as described below under mastery learning, it allows you to work at your own pace.

Required Zoom Study Groups

The class will be divided into 4 study groups that will meet via zoom twice per week. You are required to have your camera on during these meetings. During the study group meetings you can get help and demo projects. The meetings on Tuesday will be with the instructor for the course, Ron Zacharski. The meetings on the other days will be with Jema Unger, the lab aide for the course.

Required Materials

Textbook

There is no textbook for the course.

Electronics Kit

The required electronics kit is available from the bookstore. You can also get the kit from Amazon. You need

- [ELEGOO UNO Project Super Starter Kit]()   ($36 as of this writing)
- [Adafruit NeoPixel Ring - 16]() ($12 as of this writing)

Laptops

You will need a laptop (or desktop) for the course running either Microsoft Windows, Apple OSX, or Ubuntu Linux.

Grading

Grading is based on a method developed by Professor Lee Sheldon at Indiana University. It is based on obtaining experience points (XP). The number of XP determines what level you are at. You start the class at Level Zero and with 0 XP. The level you obtain at the end of the semester determines your final grade. Here is the chart:

Level	XP	Grade
Zero	0	F
One	200	D
Two	230	D+
Three	270	C-
Four	300	C
Five	330	C+
Six	370	B-
Seven	400	B
Eight	430	B+
Nine	470	A-
Ten	500	A

Here are the ways of earning XP:

Demos

The only way you gain XP is by demonstrating projects to either the instructor or one of the lab aides during the study group meetings. **You cannot demo more than 50 XP worth of projects in a week** . The last day to demo is the 30th of April.

Activities

The sole thing you will be doing is building and programming devices. There will be approximately a dozen core experiments and each experiments has a number of remixes you can do to earn XP. For example, the first task we will do is to build a device that blinks an LED light. The remixes of that task including blinking two LED lights and blinking 2 LED lights in a particular pattern. You gain XP for each of those remixes. The exact XP for each remix is shown in the experience point spreadsheet.

The final activity is a creative project, which will be explained mid-semester.

What an A means.

Here's the scoop with the grading system. You don't need to be a computer geek or math wiz to get an A in the class. It doesn't matter what your skills are nor what your interests are, you can get an A. In last 2 years in this class 50% of the people received an A. An A means that you have mastered some rudimentary programming skills. It does not mean that you will do well as a computer science major and continue to get A's in computer science courses. Typically 25-30% of the people taking the next course in the computer science sequence fail. If you are interested in majoring in computer science please talk to me sometime during the semester so I can help you determine if majoring in computer science is the right choice for you.

Avatar names, pseudonyms, noms de plume

[Avatar Submission Link](#)

On the course signup form you will be asked for your avatar name. This is the name that will appear on the Experience Point Google Spreadsheet that will be viewable by everyone in the class. If you wish to remain anonymous, don't share your avatar name with anyone. On the other hand, if you would like recognition for achieving level 10 as an example ("a big shout out to tera miner for achieving level 10"), you can share your name. The decision is yours. To further protect the anonymity of those who wish to remain anonymous, the spreadsheet will also be populated by fictitious avatar names.

Do I need to be a math wiz or computer geek to succeed in this class?

I am hoping that there will be some of you who look over the above list of activities and think, wow! that's exactly what I want to do. I expect some of you will look at that list and panic. Programming in front of a teacher–that seems scary and high stress. You may hate math and feel you have no talent there. You may feel that you are not computer geeky enough to take the class. Rest assured, this is not the case. I designed this course to enable people from a variety of backgrounds to succeed. You'll be doing the majority of your programming in class and much of that time will be working with other students. If you have problems we will solve them quickly. The computer language we will be using is easy to learn but yet very expressive.

I figure if I can program chances are good that you can program. I have a Bachelor of Fine Arts in music performance. I took zero math classes in college. Zero. Maybe you think, yeah, but you were probably a math wiz in high school. Nope. I went to an all boys trade high school with basically zero math. Regardless of your background, if you are interested in learning to program you can do it.

If you have any questions or would like more information about the course please contact me.

If you have any computer related questions, even if they are not central to the topic of this course, please come and see me.

Accommodations for students with special needs

Any student with a documented disability may receive a special accommodation to complete any requirements of this course. If you are have a disability or believe you have one you may wish to self-identify. You may do so by providing documentation to the Office of Disability Services located in Room 203 of George Washington Hall (Phone: Voice 540-654-1266, Fax: 540-654-1163). Appropriate accommodations may then be provided for you. If you have a condition that may affect your ability to exit the premises in an emergency or that may cause an emergency during class, you are encouraged to discuss this in confidence with me and/or anyone at the Office of Disability Services. This office can also answer any questions you have about the Americans with Disabilities Act (ADA).

Title IX Statement

University of Mary Washington faculty are committed to supporting students and upholding the University's Policy on Sexual and Gender Based Harassment and Other Forms of Interpersonal Violence. Under Title IX and this Policy, discrimination based upon sex or gender is prohibited. If you experience an incident of sex or gender based discrimination, we encourage you to report it. While you may talk to me, understand that as a "Responsible Employee" of the University, I MUST report to UMW's Title IX Coordinator what you share. If you wish to speak to someone confidentially, please contact the below confidential resources. They can connect you with

support services and help you explore your options. You may also seek assistance from UMW's Title IX Coordinator. Please visit http://diversity.umw.edu/title-ix/ to view UMW's Policy on Sexual and Gender Based Harassment and Other Forms of Interpersonal Violence and to find further information on support and resources.

Resources

Tiffany W. Oldfield, J.D.
Title IX Coordinator
Office of Title IX
Fairfax House
540-654-5656
toldfiel@umw.edu

Myranda Thomson
Title IX Deputy for Students
Area Coordinator
540-654-1184
mthomson@umw.edu

Confidential Resources

On Campus

Talley Center for Counselling Services
Lee Hall 106

Student Health Center
Lee Hall 112

Off-Campus

Empowerhouse
540-373-9373

RCASA
540-371-1666

Recording Policy

Classroom activities in this course may be recorded by students enrolled in the course for the personal, educational use of that student or for all students presently enrolled in the class only, and may not be further copied, distributed, published or otherwise used for any other purpose without the express written consent of the course instructor. All students are advised that classroom activities may be taped by students for this purpose. Distribution or sale of class recordings is prohibited without the written permission of the instructor and other students who are recorded. Distribution without permission is a violation of copyright law. This policy is consistent with UMW's Policy on Recording Class and Distribution of Course Materials.

Academic Integrity

I assume you are an ethical student and a person with integrity. I expect that you will follow the university honor code (see http://rosemary.umw.edu/CSHonorCode.html). Please use common sense and ask yourself what would a person with integrity do? To help you, I would like to make three comments related to this:

Plagiarism

Plagiarism means presenting some other person's work as your own. This can mean using some other person's words without acknowledging their source, or using some other person's ideas. Copying another student's work (homework or exam) is also plagiarism. Plagiarism will minimally result in a negative score for that work. For example, if the work was worth 100xp, plagiarized work will result in a score of -100.

Collusion

Collusion is unauthorized collaboration that produces work which is then presented as work completed independently by the student. Collusion includes participating in group discussions that develop solutions which everyone copies. Collusion will result in a negative score for that work.

General Education Student Learning Outcomes

- Students will demonstrate an ability to interpret quantitative/symbolic information. For example, students should be able to examine an algorithm or source code and state in English the behavior of that code.
- Students will have the ability to convert relevant information into various mathematical/analytical forms. For examples, students should be able to take a problem description and produce a working computer program.
- Students will be able to apply analytical techniques or rules to solve problems in a variety of contexts. For examples, take an idea they have for an Android application, produce a specification for how to implement that idea, and produce a working version of the program.
- Students will gain an appreciation for how analytical techniques or rules are used to address real-world problems across multiple disciplines.

Schedule

There is no schedule for the class. You complete the material at your own pace.