Summer Conference for Computer Studies Educators Schedule

You are welcome to choose any session from any "Selection" at any time.
You are not required to stay with the same column in the schedule for the entire conference.

Day 1 – Tuesday, August 13

9:00 a.m. - 10:30 a.m.	Welcome Session		
Session 1	**Selection A**	**Selection B**	**Selection C**
10:45 a.m. - 12:15 p.m.	Using AI Tools in Your Courses *(Lecture)* Grant Hutchison **MC-4041**	Philosophy of Teaching and Ethics *(Lecture)* Douglas Henrich **Math3-3027**	OMG, I Have to Teach ICD2O *(Lecture)* Rob Ceccato **Math3-3103**
12:15 p.m. - 1:15 p.m.	Lunch		
Session 2	**Selection A**	**Selection B**	**Selection C**
1:15 p.m. - 2:45 p.m.	GitHub Pages *(Lab)* Catherine Leung **MC-3005**	A Nesting Doll Approach *(Lecture)* Mike Parchimowicz **MC-4041**	Arduino Programing – From Simple Circuits to Advanced Robotics and IoT *(Lab)* John Rampelt **Math3-3103**
Session 3		**Selection B**	
3:00 p.m. - 4:30 p.m.		Online IDEs and Code Editors *(Lecture)* Lisa Rubini-LaForest **MC-4041**	

Day 2 – Wednesday, August 14

Session 4	**Selection A**	**Selection B**	**Selection C**
9:00 a.m. - 10:30 a.m.	ArenaBots *(Lab)* Nathan Robottom **MC-3103**	Using Google Colab and Google Classroom as Workflow Tools *(Lecture)* Russell Gordon **MC-4041**	Design Your Own Circuit Boards Using KiCad *(Lab)* John Rampelt **Math3-3027**
Session 5		**Selection B**	
10:45 a.m. - 12:15 p.m.		Quantum Computing *(Hands-on)* John Donohue **MC-4041**	
12:15 p.m. - 1:15 p.m.	Lunch		
Session 6	**Selection A**	**Selection B**	**Selection C**
1:15 p.m. - 4:30 p.m.	Computational Thinking Activities *(Hands-on)* Roma Uniat **Math3-3027**	KDTags and Bluetooth Communication *(Lab)* Michael Chan **MC4042**	Using PCBs and PCB Creation in TEJ3M & TEJ 4M Courses *(Lab)* Stephen Emmell **Math3-3103**

Day 3 – Thursday, August 15			
Session 7	**Selection A**	**Selection B**	**Selection C**
9:00 a.m. - 10:30 a.m.	Using p5.js to Introduce Coding *(Lab)*	Node Based Programming Using Unreal Engine *(Lecture)* *Mark Dubeau* **MC-4042**	Raspberry Pi Pico Programming *(Lab)*
Session 8		**Selection B**	
10:45 a.m. - 12:15 p.m.	*Sandy Graham* **MC-3005**	Interdepartmental Collaboration for Video Game Development in ICS4U *(Lecture)* *Andrew Seidel* **MC-4041**	*Grant Hutchison* **M3-3103**
12:15 p.m. - 1:15 p.m.	**Lunch**		
1:15 p.m. - 2:00 p.m.	**Wrap up and Resource Sharing**		

Session 1 Selection A   **Using AI Tools in Your Courses**

In this session we will explore how to teach some of the fundamental AI methods including supervised learning and unsupervised learning using various tools and techniques. We will also explore some ideas on how generative AI tools might be used in the context of designing, creating and testing applications in Grade 10-12 Computer Studies courses. A

Session 1 Selection B   **Philosophy of Teaching and Ethics**

As teachers, we are bound by the Ontario College of Teachers Code of Ethics. In addition, teachers should develop their own Philosophy of Teaching either at the beginning of their teaching career or as an experienced teacher reflecting on their teaching practice. My presentation would discuss the importance of having a Philosophy of Teaching and how a teacher can develop their own. I will also share my own Philosophy of Teaching and how it can be best presented to their Learning Community.

Session 1 Selection C   **OMG, I Have to Teach ICD20**

OMG, I have to teach ICD2O, I could really use a completed unit plan. In this session, I want to help you figure out how to structure your ICD20 course, (units, assessment, lesson's, resources, final project, etc..) by sharing with you what I did, and how I set up my ICD2O course. This will start out as a presentation, with me showing you my units, lesson's, assessments, etc.. However, I hope that after we cover my "vision" the rest of the session will be fluid and dynamic, so you can get the most help and understanding of why I set up my ICD2O course the way I did, and how you could do it for yourself.

Session 2 Selection A   **GitHub Pages**

Building and hosting client-side websites using GitHub pages.

Session 2 Selection B   **A Nesting Doll Approach**

Careful selection of projects can assist with the acquisition of programming skills. The presenter proposes a 'layering in' approach to teaching computer science/programming, where each subsequent assignment is the previous assignment with some modifications to it that incorporate the latest concepts. Upon completion, each assignment looks strangely like the previous - like a Nesting Doll.

In this approach, programming concepts and skills are first modelled through seemingly disparate, but memorable exercises with colorful names, such as "My TV is broken" which is an exercise which covering rectangular collision detection, but presented in a context that is different from that which it will appear in the major project. These exercises can be thought of as the current outermost doll in a nesting doll and are used as a mental anchor for students when assisting with problem solving. A student may ask, "Why isn't my collision detection working?" to which I might simply reply "My TV is broken."

A significant challenge in mentoring and teaching programming to a broad array of student profiles is to know how much or how little support to provide. To this end, the teacher can help to appreciate the outermost doll, but the student is responsible for putting one doll inside of the next - "how does my last project 'turn into' this next one?"

Relatively sophisticated/self-reflective ideas (such as modularity, conventions, proper form) are learned by osmosis - students are compelled to make re-useable and modular code if they experience the conveniences first-hand.

A hypothetical concept map for a one semester, Grade 11 introductory programming course, with project ideas and resources, is presented and discussed to model a way this can be done. Java Processing is used, but the projects could be adapted to a number of different languages.

Session 2 Selection C   Arduino Programming
**   – From Simple Circuits to Advanced Robotics and IoT**
Go beyond connecting Arduino to simple breadboard circuits (and then trying to figure out how to turn these into viable robot projects at the end of the semester) using open circuits specifically designed for teaching introductory programming and for building simple robots! Both the open hardware ARPS and BEAPS circuits feature common built-in I/O devices such as pushbuttons, LEDs, a beeper, and a temperature sensor. Students use these pre-wired components to rapidly progress through learning fundamental Arduino programming skills. The on-board motor driver, modular optical sensors, and integrated SONAR mount/expansion headers makes it easy to convert these circuits into simple line-following, obstacle sensing, or classroom Sumo robots, as well as IoT (Internet of Things) hubs. Participants should supply their own Arduino UNO R3 or R4 (Arduino UNO R4 Minima, or Arduino UNO R4 WiFi are recommended), or an Arduino Nano ESP32, and will assemble and program a simple classroom robot.

Session 3 Selection B   Online IDEs and Code Editors
As many school districts prioritize Chromebooks in student 1:1 programs, there is a need for an online code editor so that students may participate fully in their studies without an additional device available. In this session, we will look at the pros and cons on various online IDEs and code editors. Participants will be encouraged to use a personal email to join various online code editors and IDEs in a student role so they can judge the student experience of these websites for themselves.

Session 4 Selection A   ArenaBots
Looking for a in interesting challenge for your 12 students? ArenaBots is an arena in which students submit code for Bot programs. Bot programs must process arrays of objects with the information of other bots and bullets in the arena and must weigh competing needs of dodging, tracking, firing and looting of ammo to decide what to do each frame. Abstract Bot classes are used to show what methods are required. There is active development of a server version which allows solo bots programmed in any coding language to compete via WebSocket libraries. In Java. there are two versions, one which features solo bots and another which is more group-based, which adds Roles like healers, support and tanks. Attendees will get access to simple bot examples and training scenarios, arena code. Server code will be demonstrated.

Session 4 Selection B   Using Google Colab and Google Classroom as Workflow Tools
In this hands-on lab session, try out resources that both introduce coding concepts as topics unto themselves, and use coding to advance our students' understanding of mathematics. In particular, understand how Google Colab, the Python programming language, and the Jupyter Notebooks format can be used to make student thinking visible. The goal is to not just "do coding" with our students but to make writing code another tool that students will naturally reach for when problem solving, exploring mathematical concepts, or when looking for meaning in data. You will have time in this session to experiment with authoring your own resources as well.

Session 4 Selection C   **Design Your Own Circuit Boards Using KiCad**

Learn how to use the open-source KiCad program to create schematic diagrams and design circuit boards for your computer technology program. Participants should download and install the appropriate version of KiCad (from https://www.kicad.org) for their operating system ahead of this session, and we will interactively design a simple circuit board and generate Gerber-format output files ready to be submitted to a printed circuit board manufacturer.

Session 5 Selection B   **Quantum Computing**

Classroom-ready activities that tie into quantum information.

Session 6 Selection A   **Computational Thinking Activities**

Engage students with screen-free computational thinking activities. Discuss the invisible barriers, adaptations to activities, and connections to procedural thinking. Watch and participate in activities similar to a "robot following instructions". Set the tone for risk taking, collaboration, and learning from mistakes in your classroom culture.

Session 6 Selection B   **KDTags and Bluetooth Communication**

Wireless Controls have become an essential and household part of our daily life. In particular, bluetooth technolgy (BT) plays an important role in this area. Industries are quick to adopt and develop BT devices because the technology maintains universal standards and is easy to use. In fact, BT prototype is so simple to use that it could readily be applied in our CS classes. We could bring this state-of-art technology into our interfacing/control projects and enrich students their appreciation of programming. My presentation is composed of two parts. Part 1 introduces Bluetooth Technology, with a KDtag BT device illustrating its principles. Part 2 is a hands-on applications session. Teachers would have the chance to analyse the KDtag device in terms of communications, characteristics and control codes. At the end of the session, teachers should be comfortable to control interfacing devices (PC, arduino…) via a KDtag using Python and C.

Session 6 Selection C   **Using PCBs and PCB Creating in TEJ3M & TEJ4M Courses**

Initially, attendees will be shown a quick demo of KiCAD and what it takes to create a PCB. They will also be shown how easy it is to have those PCBs manufactured. Secondly, we will take a provided PCB as an example student project and solder them together and ideally do some simple programming examples to illustrate how they are used. This is a terrific progression I have used where students use a provided PCB in TEJ3M (soldering, programming, wiring additional circuits onto) followed up by TEJ4M where they not only learn how to create their own PCBs, but have them manufactured, and then use that same original board from TEJ3M to program the ATTiny chips on their new PCBs.

Key areas of focus:
- Illustrating the teaching of real industry software (KiCAD)
- Showcasing the viability of PCB fab use within a single semestered class
- Discussing additional uses of their project board that students receive in TEJ3M
- Basic programming pieces around using Arduino in the classroom, or the possibility of basing this idea on alternate microcontrollers (PICs, Pi Micro, Espressif, etc)

Session 7 Selection A   Using p5.js to Introduce Coding

Choosing a programming language to introduce coding can be difficult. Ideally the language syntax should be as simple as possible but also provide motivation for students to learn it. P5.js is a combination of JavaScript with a Processing library. JavaScript is the most popular language used on websites. Processing is a language designed to be easy to work with multimedia data and interact easily with the keyboard and mouse. This session will provide participants a taste of this language in action. This session will not assume any previous programming experience.

Session 7 Selection B   Node Based Programming Using Unreal Engine

The story of how my students and I published a game on the Steam marketplace.

Session 7 Selection C   Raspberry Pi Pico Programming

Hands on exploration of how to integrate Python programming and robotics into senior Computer Technology courses using Raspberry Pi Pico.

Session 7 Selection B   Interdepartmental Collaboration for Video Game Development in ICS4U

Students learn the necessary skills to work with their peers in other classes as they develop their own video game. From the media arts department creating their characters, animations, backgrounds; drama students doing voice acting; incorporating music from the instrumental students; trailer creation with the communications technology students; writer's craft to create a script for the video game; working with the business marketing team to create marketing materials; and much more! Come to this collaborative session where we brainstorm what might work in your class and see what/how my students have flourished at completing their work.