

# Berrien Springs Partnership Syllabus and Instructor Qualifications

**CLASS TITLE: Build & Program Your Own Computer**

**GRADE OR AGE LEVELS: 6<sup>th</sup> & up**

**START DATE: contact instructor for class dates      END DATE:**

**# WEEKS TOTAL: 17      WEEKS OFF: 3**

**DAY/TIME REQUIRED: Wednesday 11am-12pm**

**# HOURS (REQUIRED): 17      # HOURS (OPTIONAL): 73 (approx.)**

**TOTAL SEMESTER HOURS POSSIBLE: 90**

**LOCATION/ADDRESS: STEAMWORKS: 606 Carrol St, Suite B, Buchanan MI 49107**

**MAIN INSTRUCTOR: Rob Kerr**

**CONTACT INFORMATION:**

**phone: 269-362-5046**

**email: [rob@funlearningcompany.com](mailto:rob@funlearningcompany.com)**

**website: <https://FunLearningCompany.com/>**

**ADDITIONAL REGISTRATION AT SITE REQUIRED? NO**

**MAIN INSTRUCTOR QUALIFICATIONS:**

I am passionate about helping students to explore new things and express their creativity. I have been teaching myself for the last eight years. For the last six of those, I've been partnering with different schools through the Fun Learning Company to teach critical thinking and STEAM courses. I enjoy tinkering and making things myself, and seek to encourage the entrepreneurial spirit in others as well. I have published my own card game, as well as writing and producing an album. One of my favorite things is when I can combine multiple interests together, like when I wrote a ukulele song for my Drone Building students to help them remember yaw, pitch & roll. I love hearing about my students' interests and seeing how we can bring those into our classroom as well. I hope to continue learning from, and with, young people for years to come.

**COURSE DESCRIPTION (complete overview shown on website):**

Learning about electronic components, programming principles, and terminology through interactive projects and immersive work on the computer they build as well as videos and lessons of general concepts/examples. Some projects can be completed semi-autonomously by following instructional video lessons and asking for help when needed. Students are encouraged to follow their interests and experiment with their own programming ideas.

## SYLLABUS/OUTLINE: weekly breakdown of Project-Based Learning activities

**Week 1:** Build, connect & update the Piper computer; discuss reading blueprints and their history, main components of a computer, and how to properly setup + care for your Piper computer.

**Weeks 2-5:** Go through the Piper story mode, learning about individual electrical components such as LEDs, buttons, switches, buzzers, and breadboards and building small circuits to interact with the Minecraft world; simultaneously, get an understanding of the Minecraft world and get ideas for what is possible to program in Minecraft (everything in the Piper world, we can recreate ourselves when we start programming in Python).

**Weeks 6 & 7:** get a basic intro to high-level programming concepts like loops, if-statements, and functions, as well as some practice writing small programs in Blockly.

**Weeks 8-11:** Combine building small electrical circuits with Blockly programming as you work through challenges creating your own mini-games with Piper Code.

**Weeks 12-13:** get an introduction to Python, and programming languages in general; Try out some simple programs in Python to post to the chat in Minecraft, teleport the player around the world, and create a block.

**Weeks 14-17:** Look at different loops in python, practice with coordinates, and try out error-handling as you speed build a pyramid, create a teleportation your of your Minecraft world (including surprises like traps and super jumps), and create a secret door that only opens if the password is guessed correctly

## COURSE OBJECTIVES AND APPROXIMATE TARGET DATES:

- Circuitry and current flow – including open/closed circuits and how buttons + switches are used to close circuits.
- Debugging broken circuitry (both virtual and concrete circuits).
- Reading and following blueprints and electronics wiring diagrams.
- Electronic components and their uses, including: LEDs, buttons, switches, & buzzers.
- Definition of programming languages and basic overview of how they're executed/interpreted, as well as a brief comparison of some commonly used languages.
- Blockly programming syntax and use.
- Boolean logic and operators (including Venn Diagrams and truth tables – briefly mentioning binary data).
- Two and three-dimensional coordinate systems (including teleporting and building three-dimensional objects)
- Python syntax and use (including the difference between the shell and using a text editor).
- Debugging programs and interpreting errors.
- Programming concepts including:
  - Variables and data storage – including data types (strings, integers, floats, bools, etc.) and lists/dictionaries
  - Libraries and Classes (importing and uses, not defining)
  - Data input/output (reading in, printing, and displaying to Minecraft)
  - Repeating code with loops (for and while loops)

- Objects, functions and methods – defining and using (including passing in arguments and returning values from functions)
- If-statements and coded decision-making
- Events and event handlers
- Variable scope and naming convention
- Multi-threaded programs, timers, and callback functions

## STUDENT ASSESSMENT - what will be used to evaluate student progress and/or end of semester pass/fail status?

All classes abide by the following:

- 1) Student agrees to attend at least 80% of class sessions/lessons offered. Attendance is kept online and tracked by Partnership staff. Failure to meet 80% or be on track to meet 80% may result in program discontinuation.
- 2) The Partnership Student Assessment or Performance Form is filled out by the teacher and turned in to Partnership staff. The link to this form is found on the web page for this class. Failing marks for lack of participation, behavior issues, practice time, etc. may result in program discontinuation.

### Class-specific assessment:

Our instructor will evaluate each student using Berrien's evaluation form and passing criteria will be based solely on students attending and actively participating in the class sessions.

Additionally, students will take our course pre/post assessment in Moodle. (We can provide your teachers with non-editing accounts to see our virtual Moodle courses upon request).

## ADDITIONAL RESOURCES: (online, books, video, etc.):

Each student will have their own login with access to our virtual Moodle course, Build & Program Your Own Computer.

## CLASS POLICIES: ATTENDANCE, BEHAVIOR, WEATHER, ETC.

**Attendance:** attendance is required, and students should notify the instructor in advance of any absence.

**Behavior:** any behavior issues will first be privately brought to the attention of the parent and, should the behavior persist, to the partnership staff.

**Weather:** the classes will be cancelled on any days when Berrien Springs Public Schools are closed. We will also contact families to remind them of this in the event of a

weather-related cancellation. We will make up any canceled class meetings at a later date.