Introduction to Computer Science with MakeCode for Minecraft

About this Course

This is a semester-long course targeted at middle school grades 6-8, as an introduction to Computer Science. The course is written for teachers who may have never taught computer science before. During this course, students will explore concepts in computer science through Microsoft MakeCode and Minecraft.

Scope & Sequence

1. Introduction and About
   - About this Course
   - About the Lessons
   - About Blocks
   - About Minecraft
   - About the Authors
2. Events and Event-Driven Programming
   - Unplugged: Events and Handlers
   - Activity: Yellow Brick Road
   - Activity: Sing a Song of Sixpence
   - Activity: Last Stand
   - Independent Project
3. Coordinates
   - Unplugged: World Landmarks
   - Unplugged: Classroom Coordinates
   - Activity: Create a Compass Rose
   - Activity: Minecraft Moving Company
   - Independent Project
4. Variables
   - Unplugged: The Rhythm Robot
   - Activity: Chicken Storm
   - Activity: Arrow Counter
   - Activity: Fall is in the Air
   - Independent Project
5. Iteration
   - Unplugged: Everyday Tasks
   - Unplugged: Walk Around the House
   - Activity: Introduction to the Agent
   - Activity: Dance Dance Agent
   - Activity: Gimme Shelter!

- Independent Project: Staircases
6. Conditionals
    - Unplugged: Silent Simon
    - Unplugged: Different Conditions
    - Activity: Tree Chopper
    - Activity: All Mine!
    - Collaborative Independent Project
7. Functions and Parameters
    - Unplugged: PB & J
    - Activity: Leap of Faith Mini-Game
    - Activity: Tree Hunter
    - Independent Project
8. Arrays
    - Unplugged: Arrays in Real Life
    - Unplugged: Bubble Sort
    - Activity: We Built a Zoo
    - Activity: Warp Belt
    - Independent Project
9. Artificial Intelligence
    - Unplugged: Paper AI
    - Activity: Maze Generator
    - Activity: Maze Pathfinding
    - Collaborative Independent Project: Build an AI
10. Final Independent Project

About the Lessons

	Unplugged Activity	Birdhouse Activity	Student Project
Concepts	High	Med	Low
Skills Development	Low	High	Med
Creativity	Med	Low	High

Each lesson is divided into three types of activities:
1. Unplugged activities
2. Guided Activities or "Birdhouse" activities, and
3. An Independent project.

These activities complement each other because they reinforce different concepts, skills, and opportunities for creativity.

Unplugged Activities
Unplugged Activities are a great way to introduce new concepts in a fun way that gets students up and away from their computers and interacting with each other in person. It's important to introduce the concept first so that students have an introduction to the big ideas they will be exploring with Microsoft MakeCode and Minecraft.

Guided "Birdhouse" Activities
Birdhouse Activities are activities that provide step-by-step instructions, and every student makes the same thing. Like the birdhouses you may have made in school wood shop, they introduce new skills and provide an easy way to assess student work because you already know what it's supposed to look like at the end. Students can work through these Activities at their own pace and implement optional extensions at the end of these Activities if they find they have extra time. Birdhouse activities provide high skills development but low creativity because everyone is making the same thing. It's important not to stop after these activities. Go beyond the Birdhouse!

Independent Projects
Each lesson culminates in an independent project, an opportunity for each student to "show what you know". They don't need to go out learn new skills, necessarily, just demonstrate the skills they learned in the Birdhouse activity, in a new context. Some of the independent projects are paired independent projects, in which students are encouraged to work with a partner on building something together. Creative projects enable students to apply new skills and concept knowledge in a unique context and provide lots of opportunity for creativity. Try to build in opportunities for students to share their projects with each other, or with the class!

About Blocks

In 1975, Seymour Papert of the MIT Media Lab created a beginners' programming language called LOGO. He developed it based on research that showed that playing with blocks of code was a particularly effective way to teach programming concepts. Papert coined the term "constructionism" to describe the way that learners construct new knowledge by building on established knowledge. The blocks in MakeCode, and the blocks in Minecraft, are themselves models for the way that new learning happens through the application of concepts in an open-ended learning environment. Block-based programming languages such as Scratch and MakeCode build on Papert's research and are a great way for students to start learning about coding concepts without having to worry about syntax.

If you are brand new to Minecraft, or just need a refresher, or if you are curious about how a game can be used to teach students, continue reading. Otherwise, feel free to skip ahead to Chapter 2, and start coding!

About Minecraft

What is Minecraft? Simply put, Minecraft is the biggest sandbox you have ever played in. The landscape stretches infinitely in all directions, as far as the eye can see. It extends up into the hills and mountains, and down into dark caverns filled with lava and treasure. Just like a sandbox, there are no specific goals in Minecraft, no set objectives to fill. The fun comes from setting your own goals and deciding how you want to play on a given day. Maybe you will head into the jungle in search of a jungle temple. Perhaps you will build your dream treehouse with a perfect view of the ocean, or mine deep underground in search of emeralds and diamonds. Whenever one task becomes too difficult, or boring, or when you get stuck and need some help or inspiration, you can just choose to do something else. There is always something engaging to do, and no matter what you choose to do, you will always learn more about the world around you and be thinking about better and more efficient ways to accomplish what you want to do.

At first look, Minecraft is not visually impressive – new players are often struck by how "blocky" everything looks. But after playing the game for a while, they become immersed in the game environment, and they realize the complexity of the game world. It's the situations and the challenges that make Minecraft seem real, not the graphics or the animation. In fact, the relatively crude graphics give it a cartoonish feel that is charming and disarming in its own way, keeping the feel of the game light.

Teaching with Minecraft is like teaching with blocks. The blocks are simply an engaging tool for the lessons you build around them. You could allow your kids to play with the blocks freely, see what they come up with, and share and celebrate their creations. You could use the blocks to build an imaginary town, and have a discussion about where the roads should go, what the buildings should be called, and whether or not you will need a jail.

Coding in Minecraft is like giving yourself superpowers. Many students are already accustomed to how Minecraft works normally. The prospect of using code to automate tedious tasks such as mining or chopping trees is immensely motivating to Minecraft players and the ability to immediately see the results of the code you wrote in the Minecraft world is incredibly powerful. Students can use a flat world to start out with, as a "sandbox" for their code, and as they learn and experiment, they will be able to see the progress of their learning visually in their world as a series of different physical projects and constructions.

For an in-depth introduction to Minecraft, you can download and install Minecraft Education Edition and follow the tutorial here: https://education.minecraft.net/worlds/tutorial-world/.

For more information on how to download and install Minecraft and Microsoft MakeCode, see the Setup Directions here: https://minecraft.makecode.com/setup

About the Authors

Douglas Kiang

Douglas Kiang is a speaker, teacher, and workshop presenter with twenty-eight years of teaching experience in independent schools at every grade level. He currently teaches high school computer science at Punahou School in Honolulu, Hawaii. Douglas holds a master's degree in Technology, Innovation, and Education from Harvard. Douglas is a Microsoft Innovative Educator and a Minecraft Global Mentor. You can follow him on Twitter at @dkiang.

Mary Kiang

Mary Kiang has been teaching for over twenty-five years at elementary, middle, and high school levels. She also developed curriculum in the Education Department of the Museum of Science in Boston. She currently teaches sixth grade Math/Science at Punahou School. Mary is a former programmer for Houghton Mifflin and Dun & Bradstreet and holds a Master's degree in Elementary Education from Simmons College. Mary is a Minecraft Global Mentor as well as the founder of GO Code!, an organization that supports girls and young women in exploring coding and STEM. You can follow her on Twitter at @maryrkiang.