

# Let's Learn Computer Science 3



Created by Code.org on June 13, 2015; last modified by Tom Seaman on June 13, 2015

## Unit Summary

This is Course 3 of a three-course set developed by Code.org intended to give elementary school students an engaging hands-on introduction to computer science. Course 3 is for students who can already read, and delves deeper into the computer programming concepts introduced in Courses 1 and 2. Course 1 is aimed at early readers, generally in the K-2 grades. Teachers of students who can read and have no prior programming experience may elect to begin with Course 2. Course 2 is a prerequisite for Course 3. Each course employs a spiraling curriculum, where concepts are introduced and later revisited, delving deeper.

All three courses in the set utilize a blend of brief (1-2 minute) video introductions, teacher-led classroom activities, and hands-on exercises where students are expected to use the programming concepts to solve problems and create things, such as an interactive game, a drawing, or a story. Students can share their creations with their friends. Each course comprises 18 or so lessons, and each lesson requires roughly 25 to 45 minutes to complete. The lessons can be introduced by the teacher with the included teacher notes and prep videos. Also included are assessment plans and mapping to USA and international standards.

Course 3 introduces students to a few key concepts in computer programming, such as variables, conditionals, and nested loops. Students will gain experience with computational thinking ("thinking tricks"), reusing functions (passing variables to functions as opposed to rewriting the function all over again with each use), creating algorithms to accomplish specific tasks, and solving problems by breaking them into smaller pieces. The students work in small groups and learn to collaborate to solve challenging problems. The course includes video interviews with professional engineers who discuss topics like internet transmission methods and the benefits of crowdsourcing in language suitable for elementary school students. It offers a lesson on the responsibilities we all share as digital citizens. The course culminates with a play lab, in which students create their own interactive game, and an art activity, where they create a program to draw patterns.

## At a Glance

- **Grade:** K-2, 3-5
- **Subjects:** Math, Arts, English Language Arts
- **Topics:** Computer Science, Engineering, Design
- **Higher-Order Thinking Skills:** Creativity, Collaboration, Persistence, Problem Solving

- **Key Learnings:** Computer programming concepts, vocabulary, collaboration
- **Content Type:** Unit Plan
- **Time Needed:** 21 lessons, each 25-45 minutes. May be done on consecutive days or over 21 week period
- **Prerequisites:** Ability to read
- **License:** Read about the license and what you can do with this material [here](#).

### Learning Outcomes

- Students should gain an understanding of basic computer programming concepts, such variables, conditionals, nested loops, reusable functions.
- Students should gain experience with computational thinking, creating algorithms to complete tasks, breaking down complex problems into smaller problems, and collaborating with others to achieve a better result than one could on one's own.
- Students should learn problem solving and techniques for persevering through difficult challenges.
- Students should learn about how the internet works, how crowdsourcing works, and the responsibilities of digital citizenship.

### Things You Need

Computer with internet connection

This course requires students to use a computer or tablet with an internet connection. Students may work independently on the hands-on activities, but we recommend they work in pairs or threesomes to learn to collaborate to solve the challenges. The course utilizes YouTube to embed the instructional videos. If YouTube is blocked at your school, Code.org will attempt to play the video content through its own hosted non-YouTube video player. Additional IT help [here](#).

Link to the Unit Plan: <https://studio.code.org/s/course3>

Students who have completed course 2 will take their skills further to solve more complex problems, ultimately creating an interactive game to share.