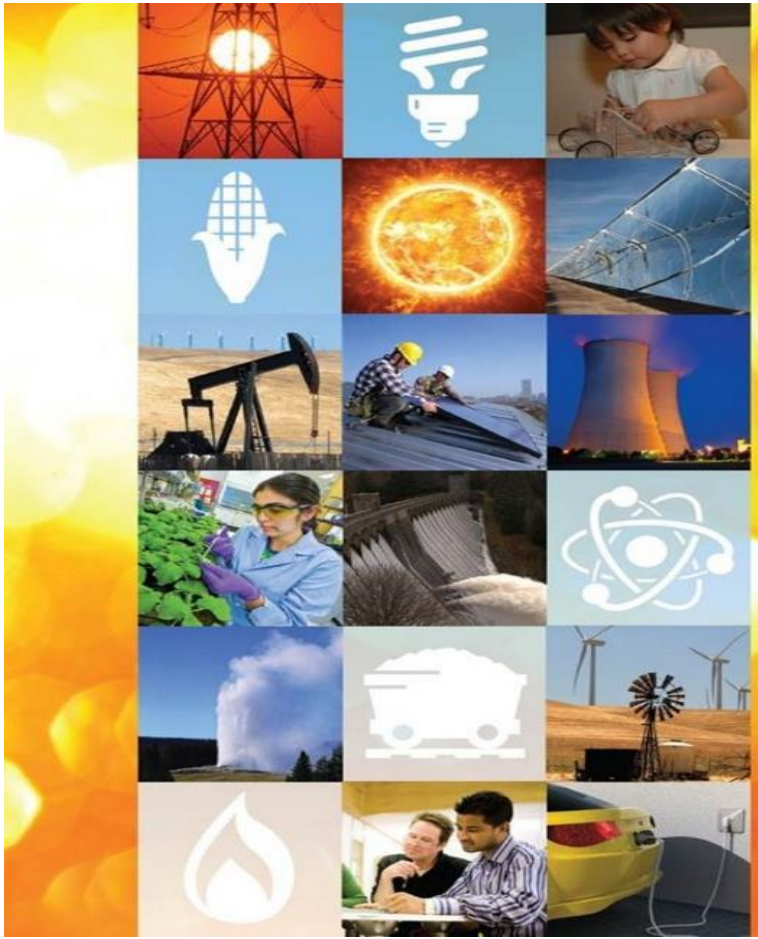
Computer Science in STEM Education

Laura T. Angle
Code.org Affiliate Trainer K-5
angle@pd.code.org

STEMpd.education/energy

In Partnership with Code.org®



Energy Literacy

Essential Principles and Fundamental Concepts for Energy Education

A Framework for Energy Education for Learners of All Ages


STEM Spark
U.S. Department of Energy

Welcome

77,630 Hour of Code events around the world

Try an Hour of Code

103,090,225 served

Anybody can learn.

Start

Beyond an Hour of Code

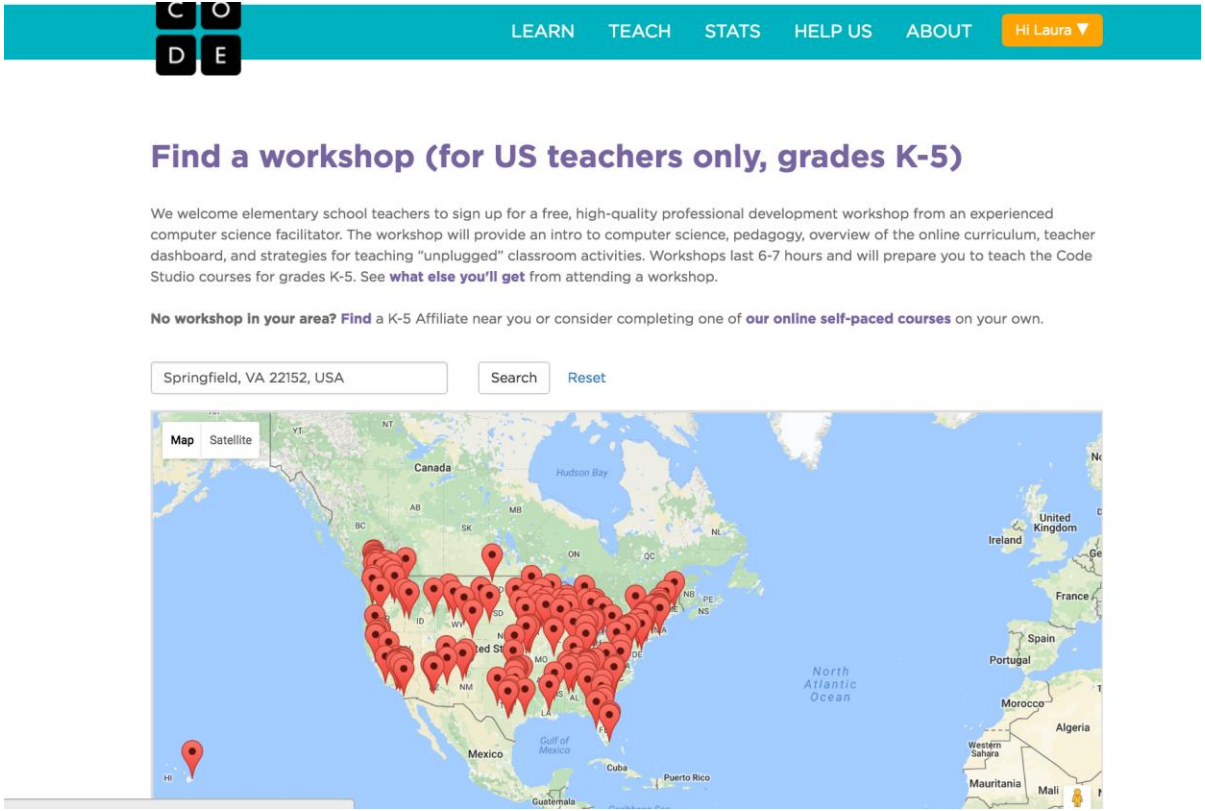
Who is Code.org?

- A public 501c3 nonprofit dedicated to bringing computer science to every school, and increasing participation by women and underrepresented students of color.
- Producer of 100 hours of online curriculum taught in 100,000 schools worldwide.
- The organizer of the Hour of Code campaign.
- Trained 35,000 teachers in 2 years
- Led campaign to create K-12 Computer Science framework.

Beyond the Hour of Code

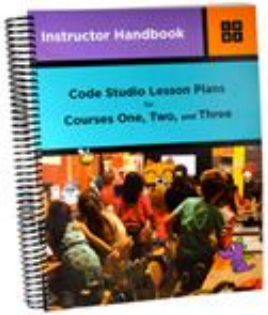
- Introducing Code Studio: Code.org's free, open-source, online curriculum.
- Four courses for grades K-5 (Over 100 hours of lessons)
- Include self guided videos with lectures by Bill Gates, Mark Zuckerburg.
- Teacher dashboard makes it easy for teachers to view lesson plans, create student accounts and monitor student progress.
- Align and reinforce concepts and skills taught in Math, English Language Arts, and Science standards.

Find a Workshop Near You



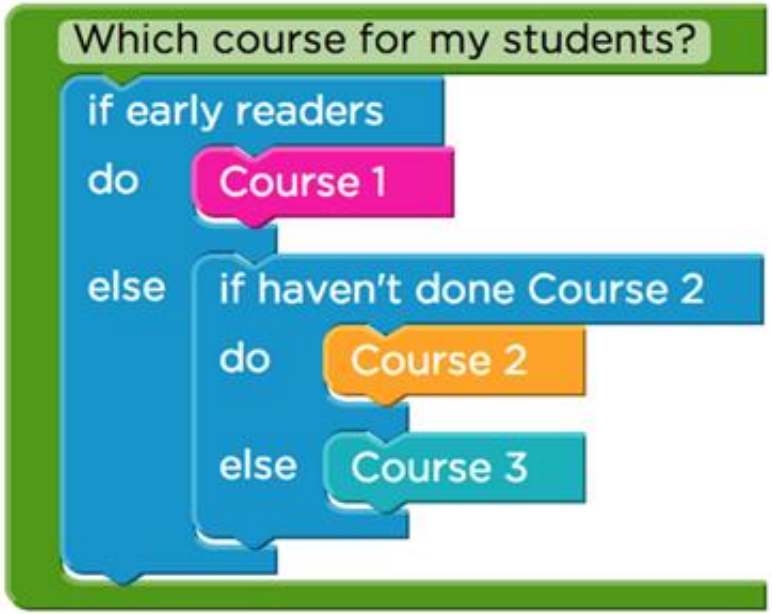
What you get in Code.org workshops

- In-person instruction
- Printed curriculum guide
- Certificate of completion
- Classroom supplies for the unplugged lessons
- Fun Code.org swag



Code Studio Curriculum

In Partnership with Code.org



@TeachCode | #CodeorgPD

Code Studio Course Overview

All ages

Course 1

Course 1 is designed for early readers.

Ages 4+ (pre-readers)

Course 2

Course 2 is designed for students who can read.

Ages 6+ (reading required)

Course 3

Course 3 is a follow-up to Course 2.

Ages 8+ (after Course 2)

Course 4

Students taking Course 4 should have already taken Courses 2 and 3.

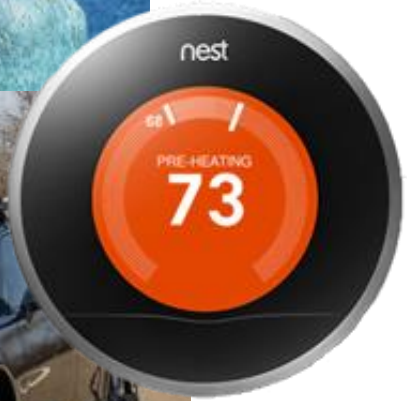
Ages 10+ (after Course 3)

- Any computer or tablet
- “Unplugged” lessons

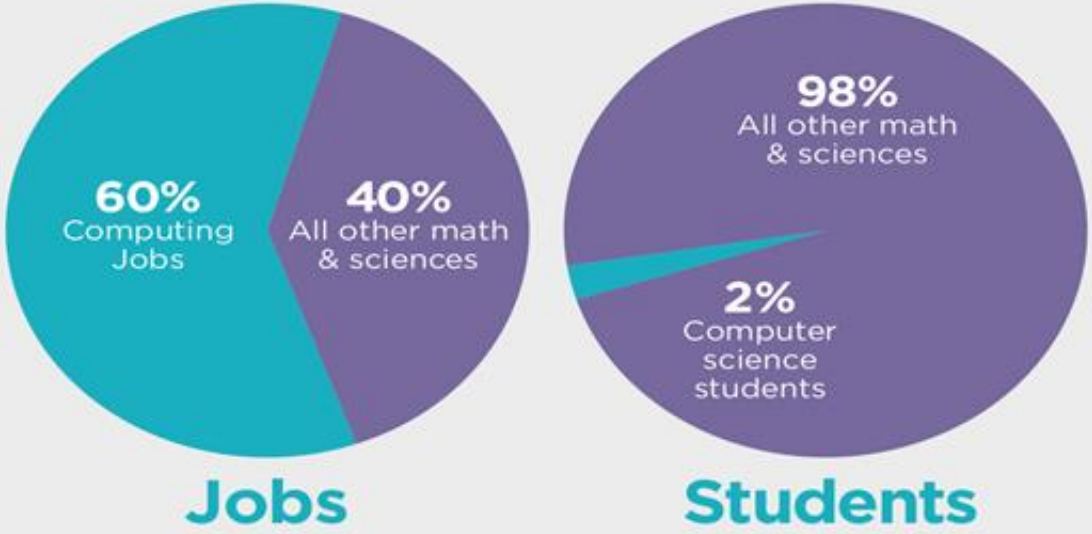
Discussion Norms

- Stay engaged.
- Speak your truth.
- Experience discomfort.
- Expect and accept non-closure.
- Embrace the “struggle”

Technology affects every field

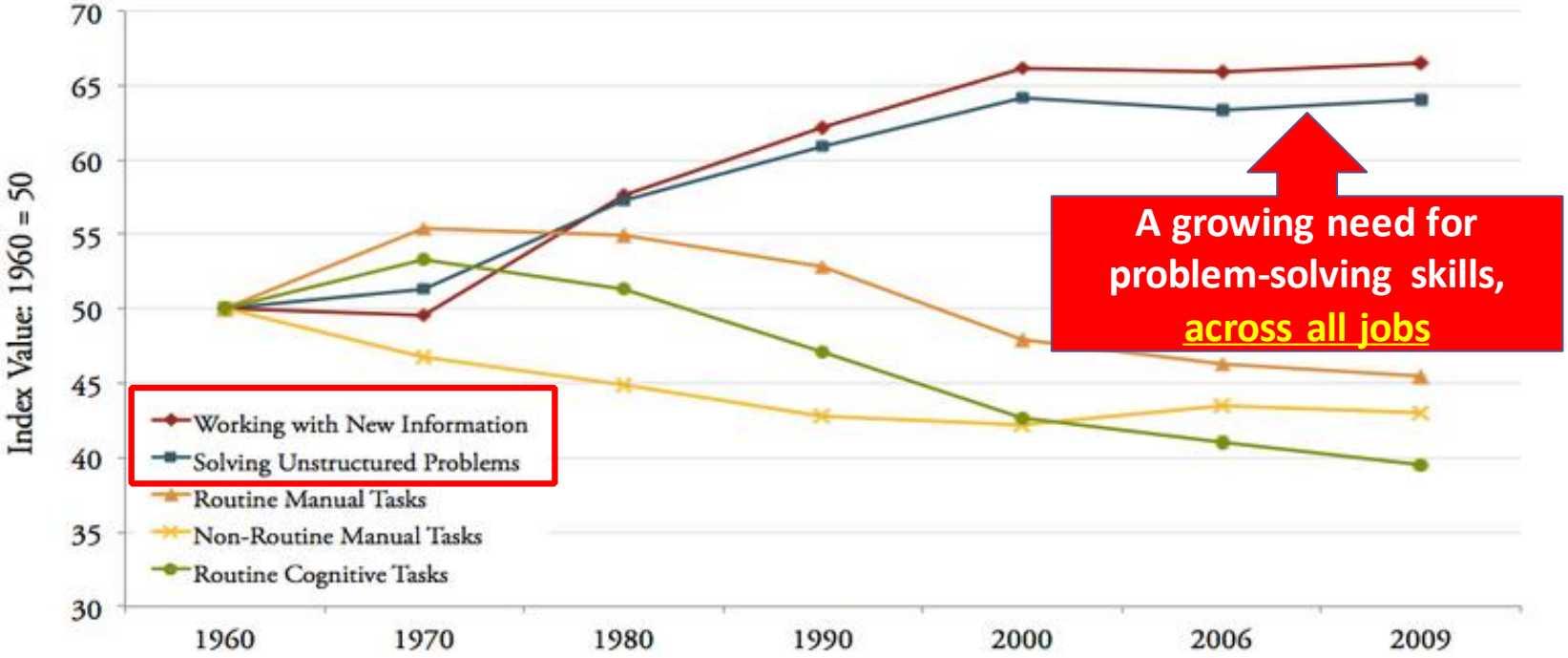


The job/student gap in computer science



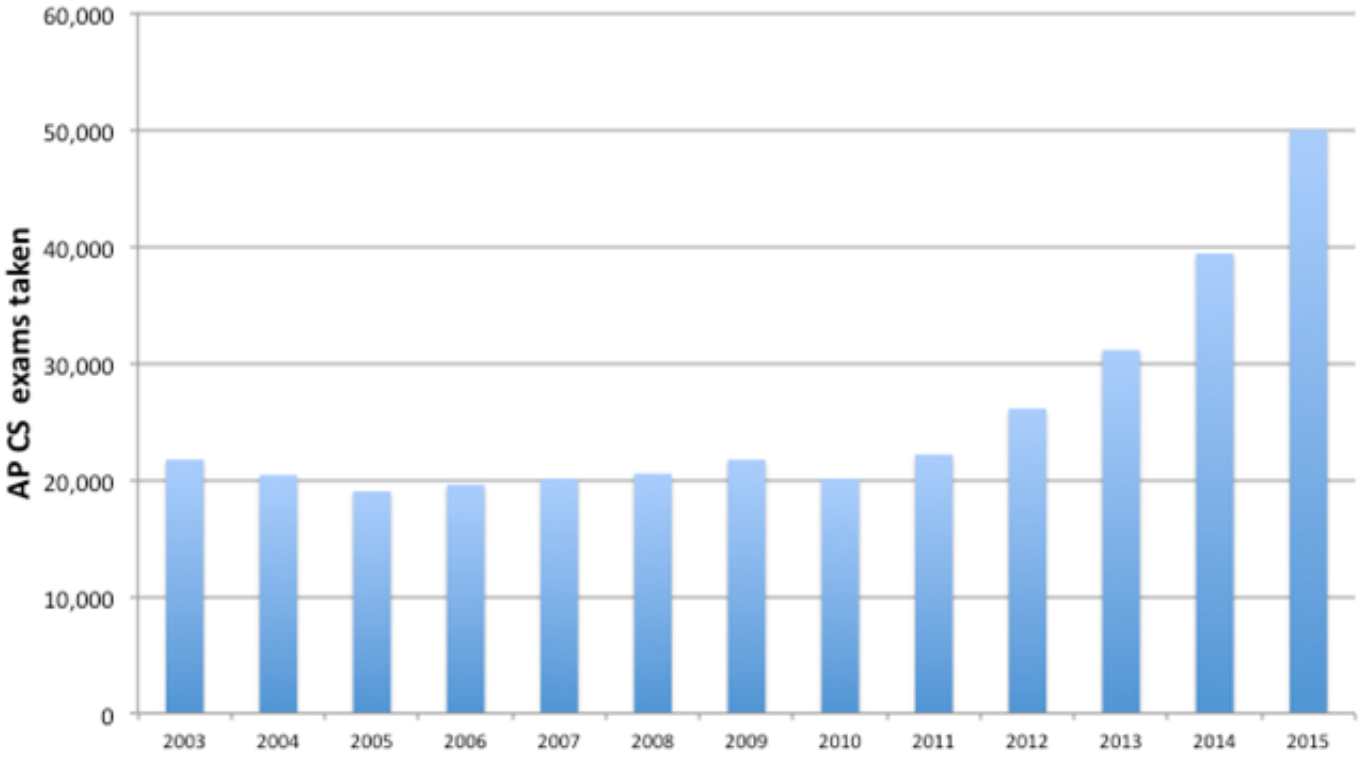
Less than 2.4% of college students graduate with a degree in computer science. And the **numbers have dropped since** last decade.

Figure 3: Index of Changing Work Tasks in the U.S. Economy 1960-2009[21]



Source: Dancing with Robots - Human Skills for Computerized Work, Levy and Murnane, 2013

Computer Science is the FASTEST growing in the 2010's

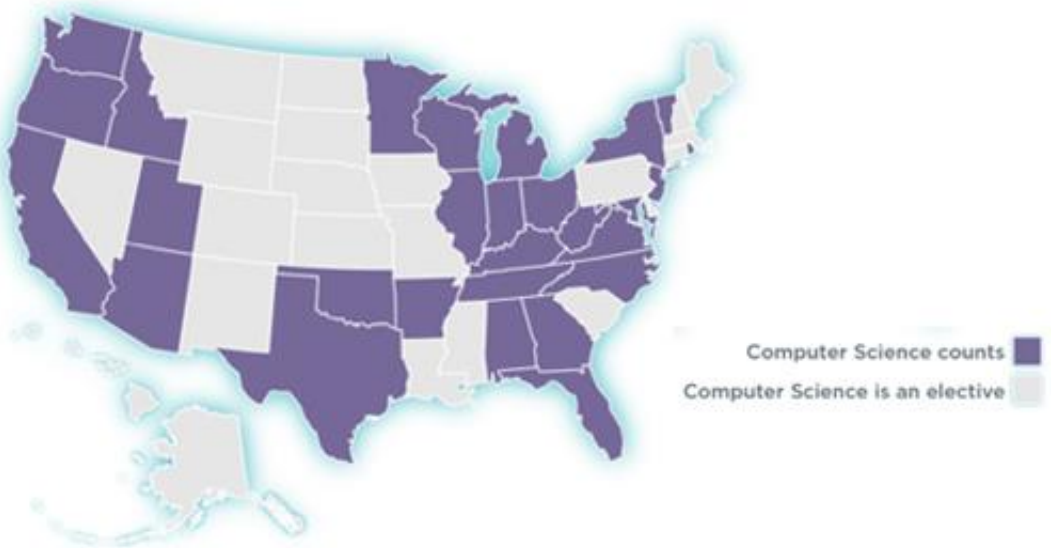


Why CS?

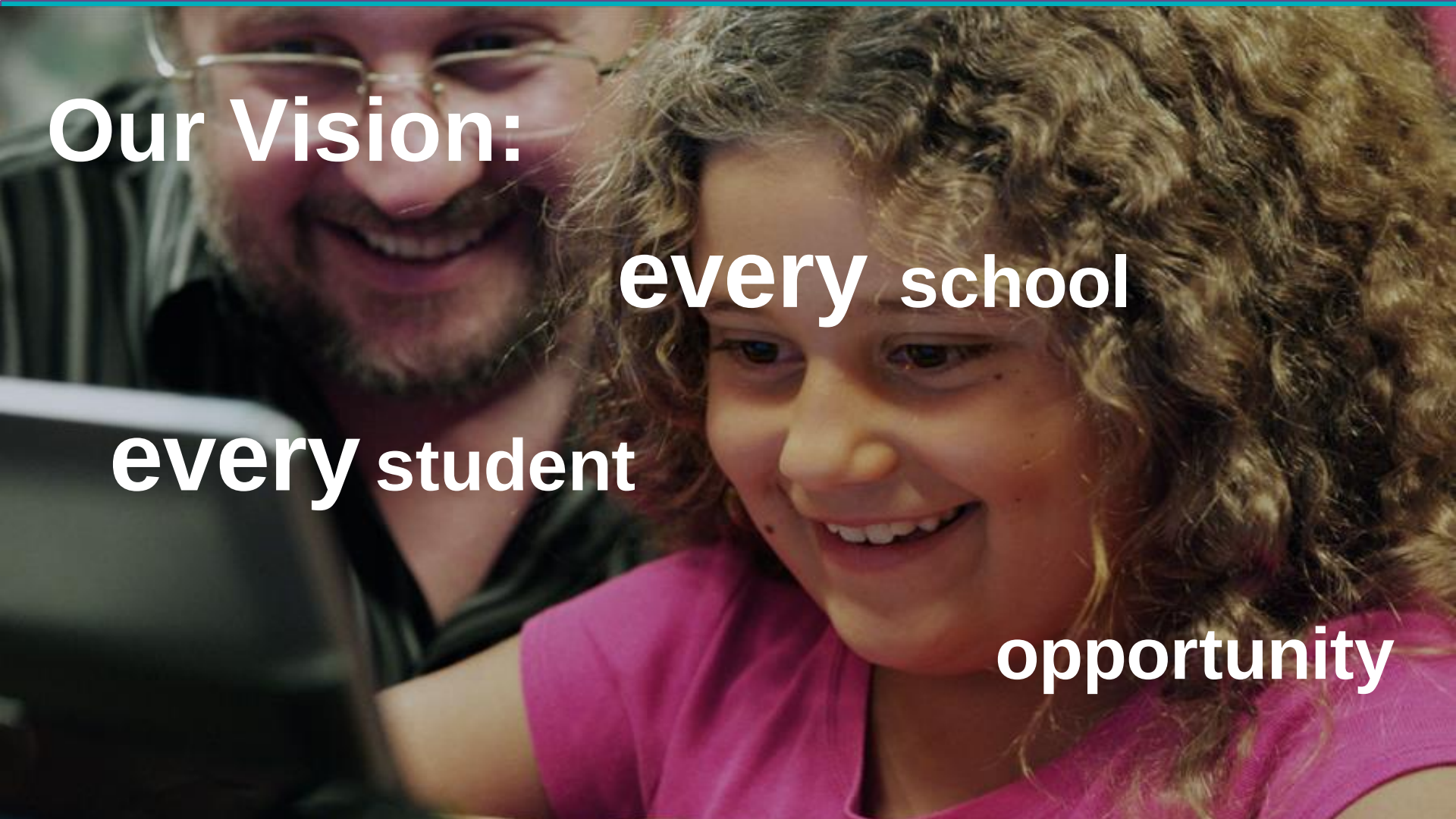
In Partnership with Code.org

CS can count for graduation in 28 states + DC

In 28 states plus DC, computer science can count towards high school graduation math or science requirements - **up from 12 states in 2013.**



Source
Data


Our Vision:
every school
every student
opportunity

Computational Thinking Practices

- Creativity
- Collaboration
- Communication
- Persistence
- Problem Solving

STEM Thinking Skills & Coding

***Science* – Scientific Method - ASK**

***Technology* – Computational Thinking - CODE**

***Engineering* – Engineering Design Process - MAKE**

***Math* – Number Theory - PATTERNS**

Code.org *Teaches logical Computational Thinking skills which works as foundation in all programming languages. C, Basic, Python, Java, etc.. And is similar to logical reasoning needed in language arts.*

Coding, Computer Science & Engineering

Coding - The set of instructions you give to a computer

Computer Science – Scientific method of inquiry on how the internet works and it's societal impacts, how technology can used to solve problems and how to prepare and break down problems into solutions using coding.

Software Engineering- Teaches engineering design process and solving computational problems using time, money and resources as a constraint.

Coding Languages

Text based - Javascript, Python, HTML, etc..

Visual based- “Blockly” - visual blocks that you drag and drop to write program.

*Easier to learn programming with visual languages because you are up against two things..

1. Learning the logic on how to write programs and
2. the syntax of writing programs (list of instructions).

Strategies for teaching CS

In Partnership with Code.org

- Ditch uniformity
- Frequent breaks
- Collaborate
- Don’t be a know it all.

Pair Programming

Vocab Shmocab

Here you go - Clap 3 times

Clap

Clap

Clap

Behind Head

Waist

Do twice : Behind Head, Waist

Behind Head

Waist

Clap 3 times

Left Up

Right Up

Do Twice : Left hand Up, Right Hand Up

Left Up

Right Up

Clap 3 times

Second round…

3x

Clap

Second round…

2x

Behind Head

Waist

Second round…

3x

Clap

Second round…

2x

Left Up

Right Up

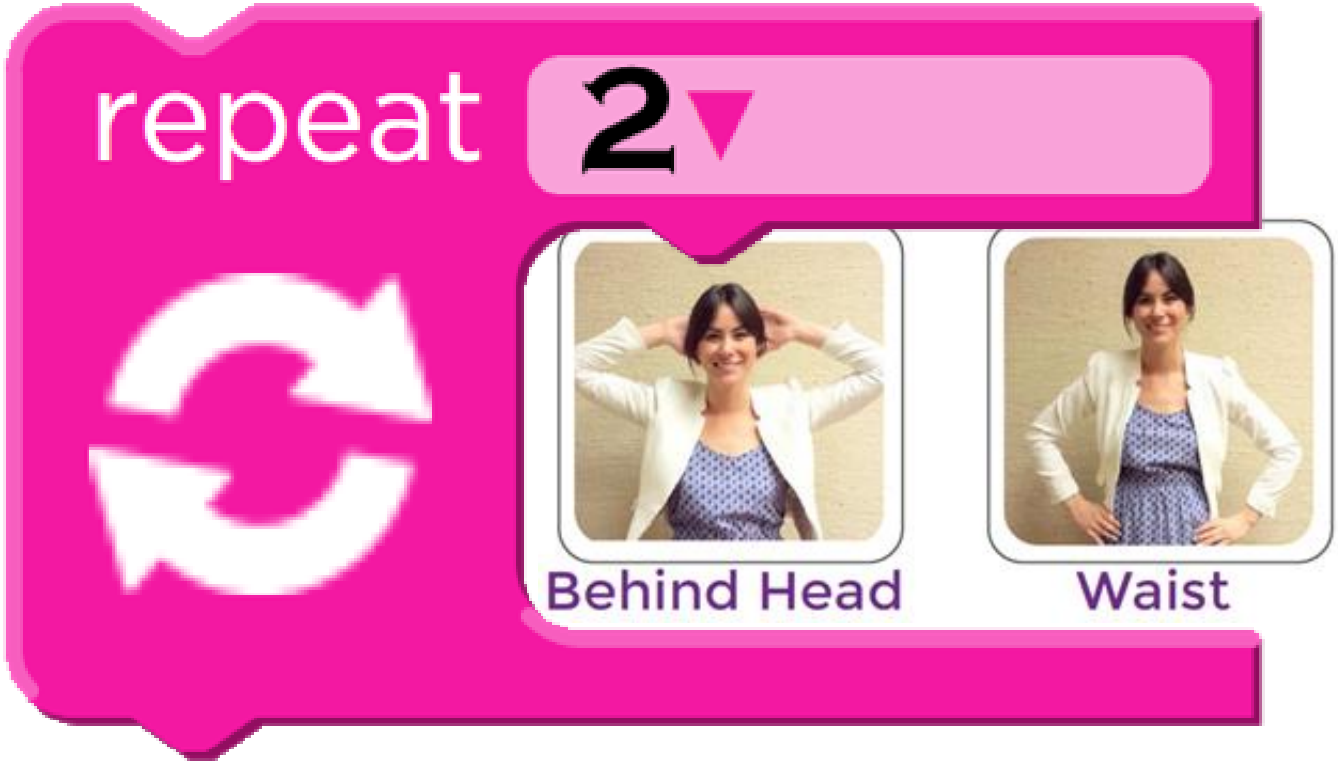
Second round…

3x

Clap

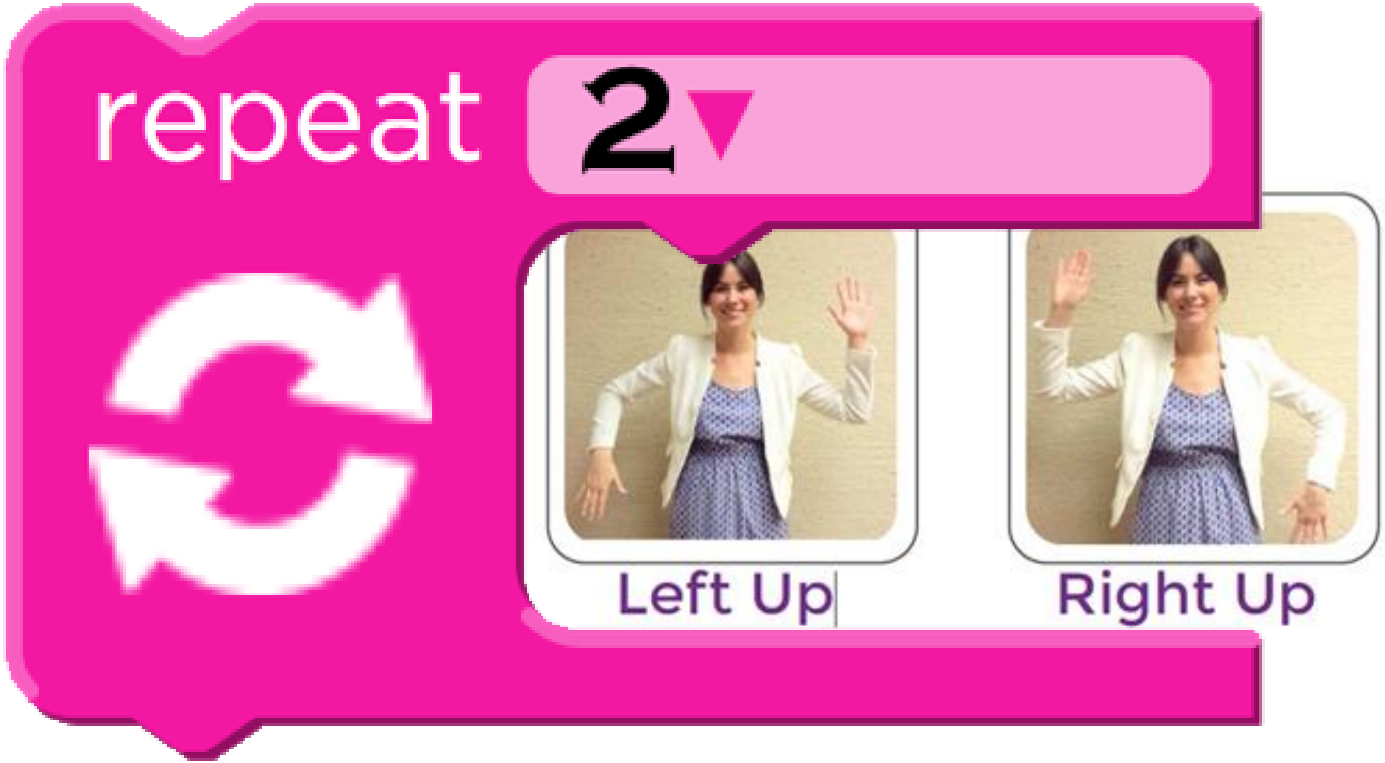
Third round…

Third round…



Third round…

Third round…



Third round…

Belly Laugh

Teacher Dashboard

In Partnership with Code.org

Teacher Home Page

Student Accounts and Progress

Your Course Progress

Lesson Plans and Resources

Help and Community

Professional Development

Computer Science Guest Speakers

@TeachCode | #CodeorgPD

Thank You!!!