Supporting the intersections between Computer Science and the NGSS.

Josh Caldwell, Code.org
Jennifer Childress, Achieve
Irene Lee, Santa Fe Institute

NSTA STEM Forum & Expo
May 22, 2015 Minneapolis, MN

Outline of today's presentation

Part I: Computer Science

- Part Ia: The Need for Computer Science Education
- Part Ib: Computer Science across the curriculum

Part II: Next Generation Science Standards

- Part IIa: Science and Engineering Practices
- Part IIb: Computational Thinking

Part III: Intersections

- Part IIIa: STEM workforce needs
- Part IIIb: Features to look for integrations
- Part IIIc: Challenges and opportunities presented

Definitions and Scope

Computer science is an academic discipline that encompasses the study of computers and algorithmic processes, including their principles, their hardware and software designs, their applications, and their impact on society (CSTA, 2011)

Computational science is a combination of computer science, science, and mathematics. Seen as a third leg of science, joining theoretical and experimental science, computational science is made possible with the advent of powerful computers.

Computational thinking is the human ability to formulate problems so that their solutions can be represented as computational steps or algorithms to be carried out by a computer.

Supporting the intersections between Computer Science and the NGSS.

Josh Caldwell, Code.org


Our Vision:
every school
every student
opportunity

Who is Code.org

Non-Profit organization that launched in 2013
with a video

Dedicated to expanding participation in computer science
increasing participation of females and
underrepresented students of color

Have a vision that every student in every school should have the opportunity to learn computer science

Have a belief that computer science and computer programming should be part of the core curriculum alongside other STEM courses

Computers are changing everything, yet most schools don’t teach computer science

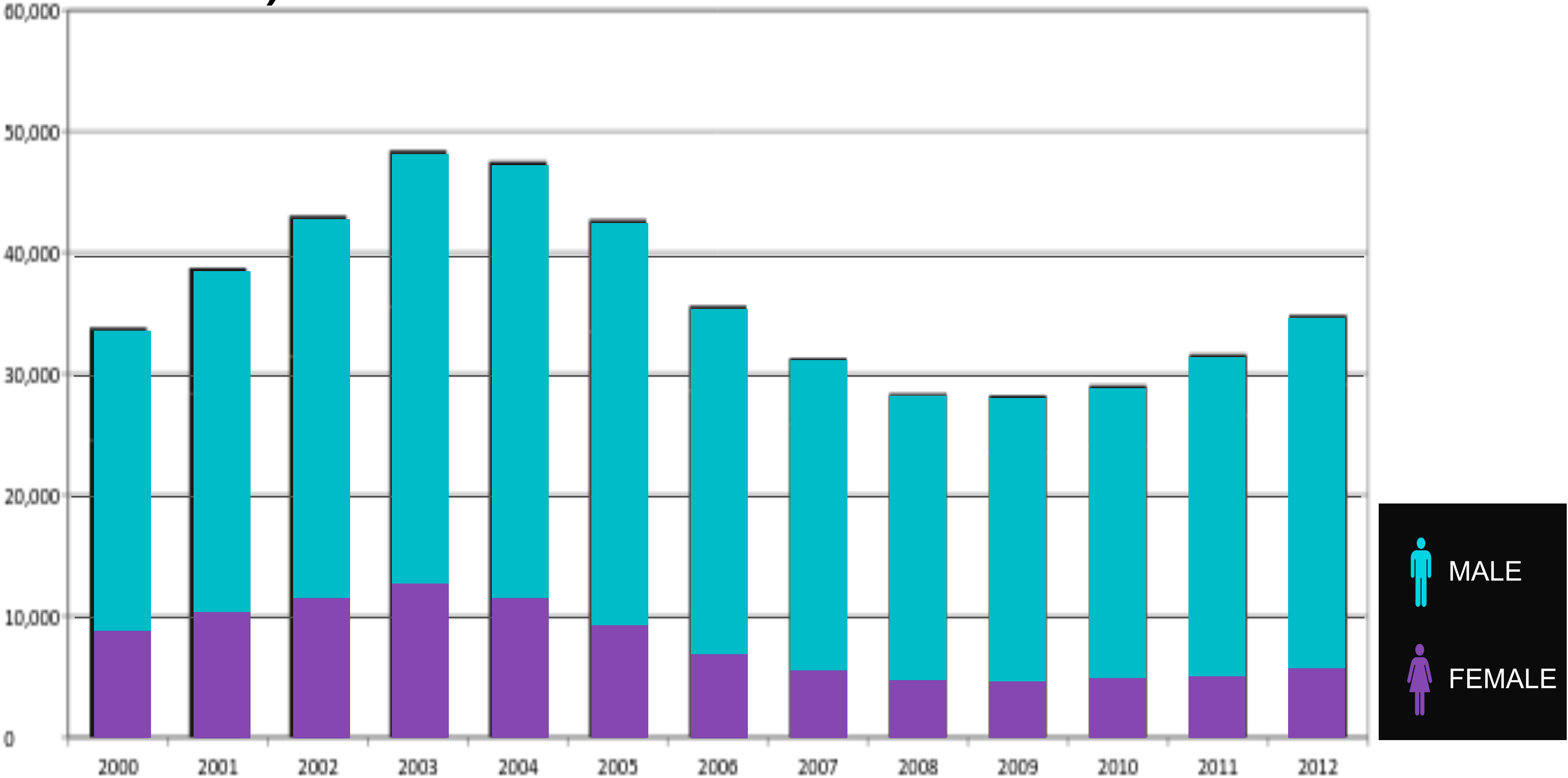
Our kids should be learning ~~to code~~.

Our kids should be learning computer science.

Computer science education is on the ~~rise~~.

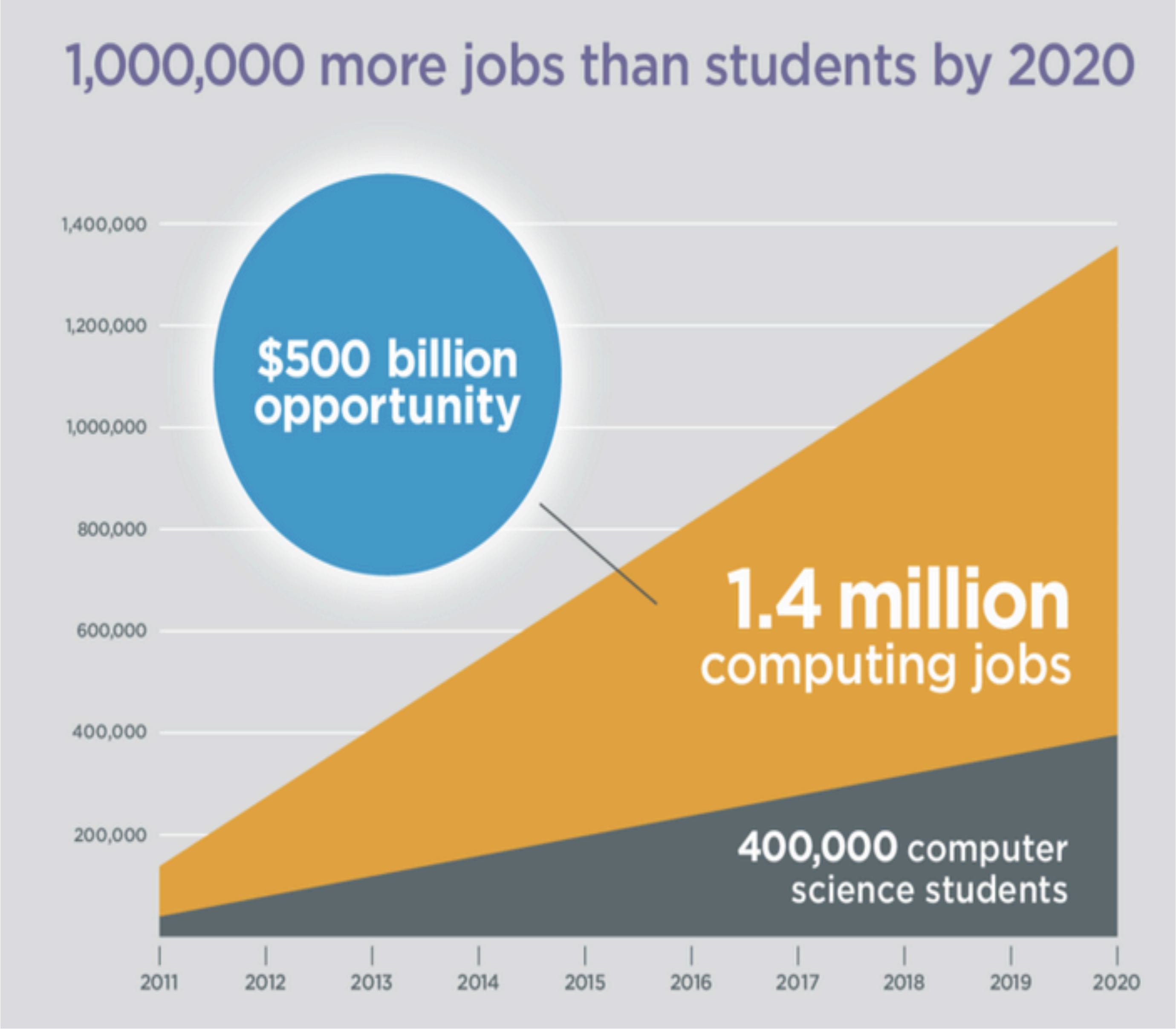
Computer science education is on the Recovery from a 10-year decline.

Fewer computer science students than 10 years ago (and half as many women)

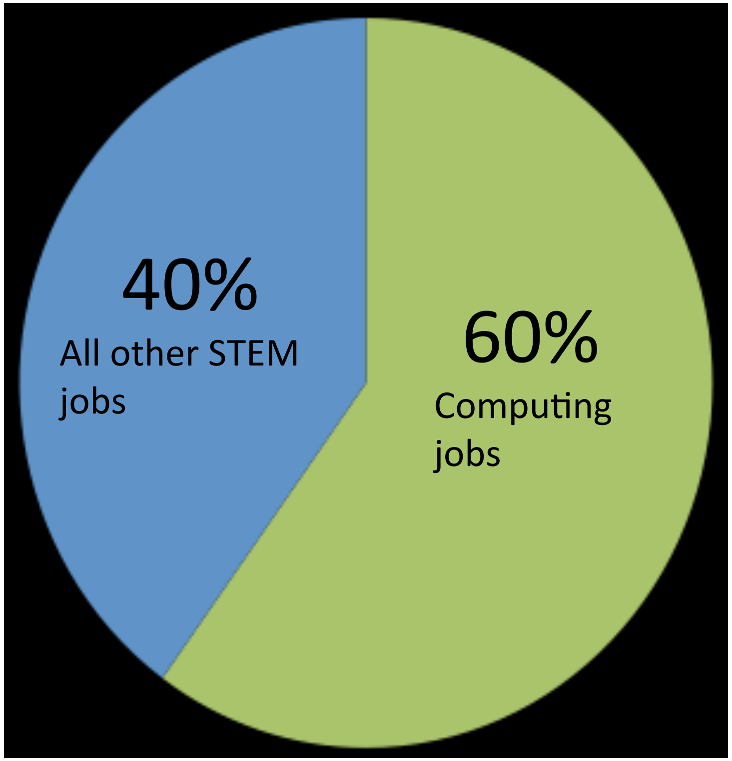


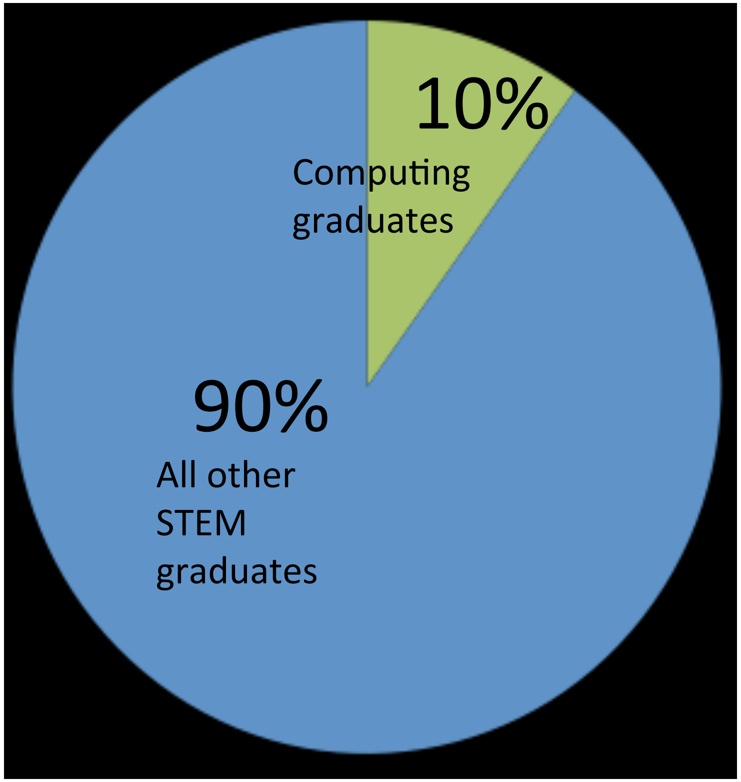
~~The tech~~ industry is desperately trying to hire computer programmers.

Every industry is desperately trying to hire computer programmers.


1,000,000 more jobs than students by 2020
1,400,000
1,200,000
1,000,000
800,000
600,000
400,000
200,000
$500 billion opportunity
1.4 million computing jobs
400,000 computer science students
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020

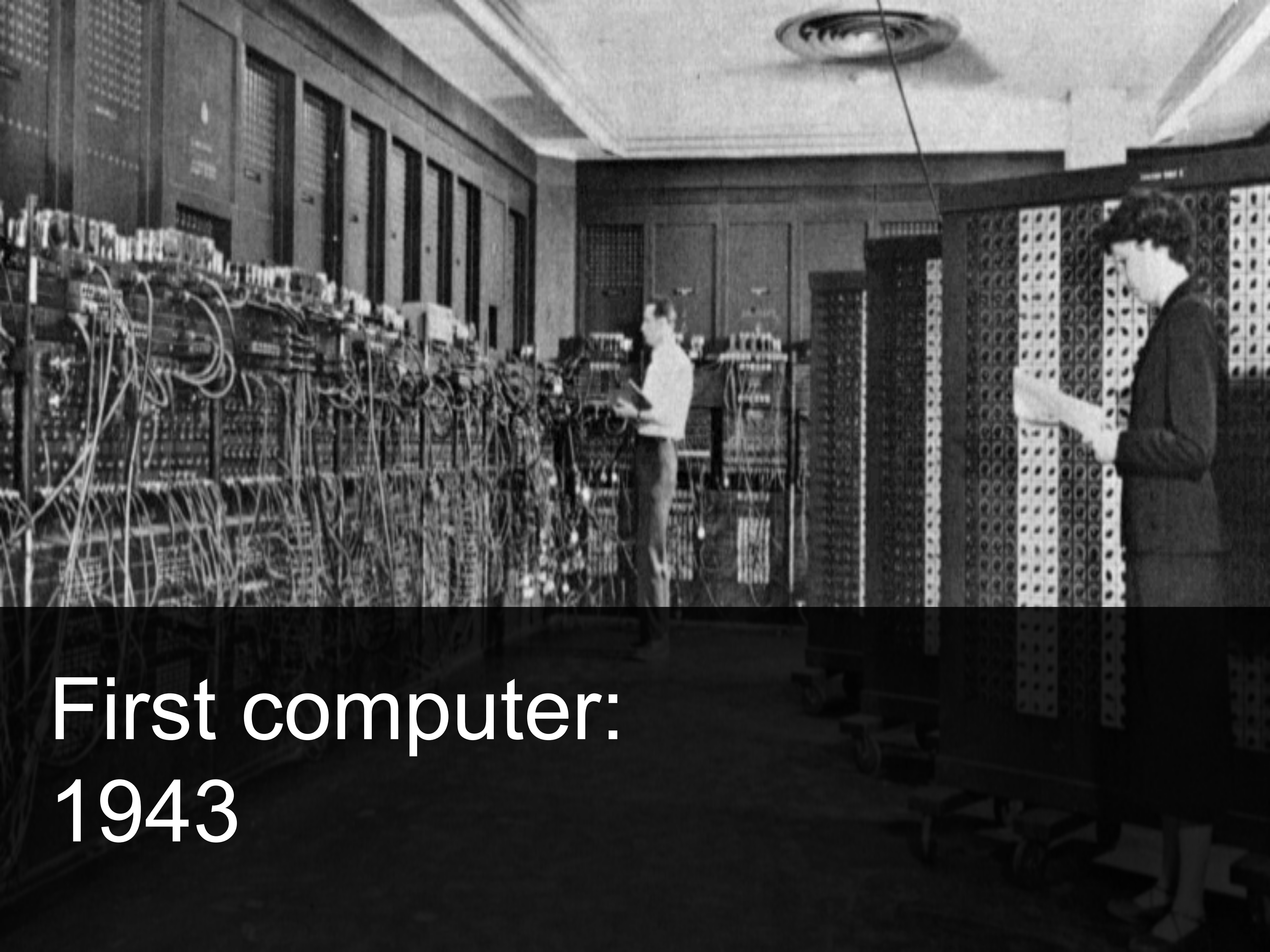
The "STEM" problem is in CS





Computer science is about ~~technology.~~

Computer science is about
logic, problem solving,
and creativity.


First computer:
1943


Ada
Lovelace
First computer: 1943
First computer program: 1843

Computer science is ~~vocational.~~

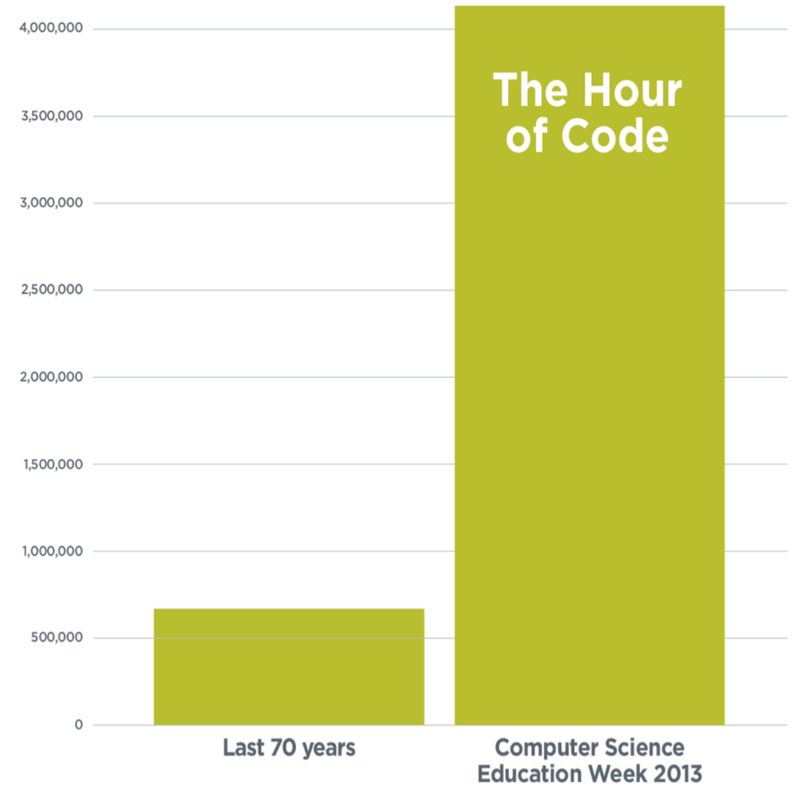
Computer science is foundational.

Can the public education system evolve?

Introducing the Hour of Code

A grassroots movement fueled by 200 partners, 100,000 teachers, in all 196 countries

In just 1 week:



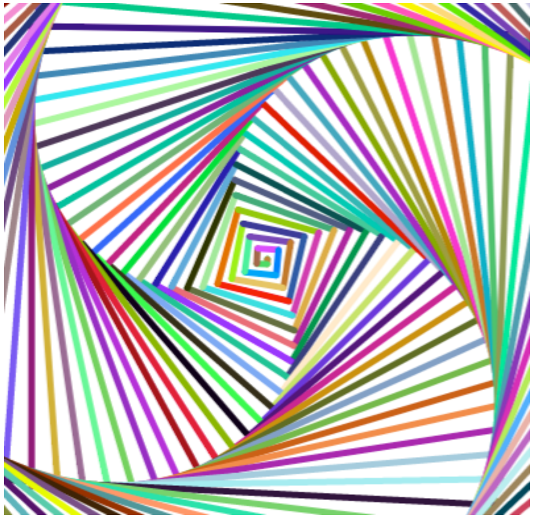
More girls participated in computer science

than in the last 70 years.

Number of Students:

What can you learn in an hour?




What’s after the #HourOfCode?

Code.org's 3 Pillars

Advocate	Celebrate	Educate
National and Regional Political Advocacy Make CS count	Videos Celebrities The Hour of Code	Elementary School • 4 20hr courses Middle School • CS in Algebra • CS in Science High School • Exploring CS • AP CS Principals

Code.org's Middle School Programs

Computer Science in Algebra

- Developed with Bootstrap
- Block-based functional programming
- Integrated into Pre-Algebra or Algebra I
- Roughly 20 hours of curriculum

Computer Science in Science

- Developed by Project GUTS
- Block-based modeling and simulation programming
- Integrated in Earth, Life, or Physical Science
- Roughly 20 hours of curriculum

Blended Professional Development

- Spring: Pre-workshop online prep (~2 hours)
- Summer: In-person workshop (3 days)
- Summer: Online reflection and planning (~8 hours)
- School Year: Online and In-person follow up

Workshop costs and
teacher stipends covered by Code.org

Science PDs with openings this Summer!

Ohio, LA & Orange County CA, Seattle, Atlanta, Houston, Charles County MD.

Supporting the intersections between Computer Science and the NGSS.

Jennifer Childress, Achieve

NEXT GENERATION

SCIENCE

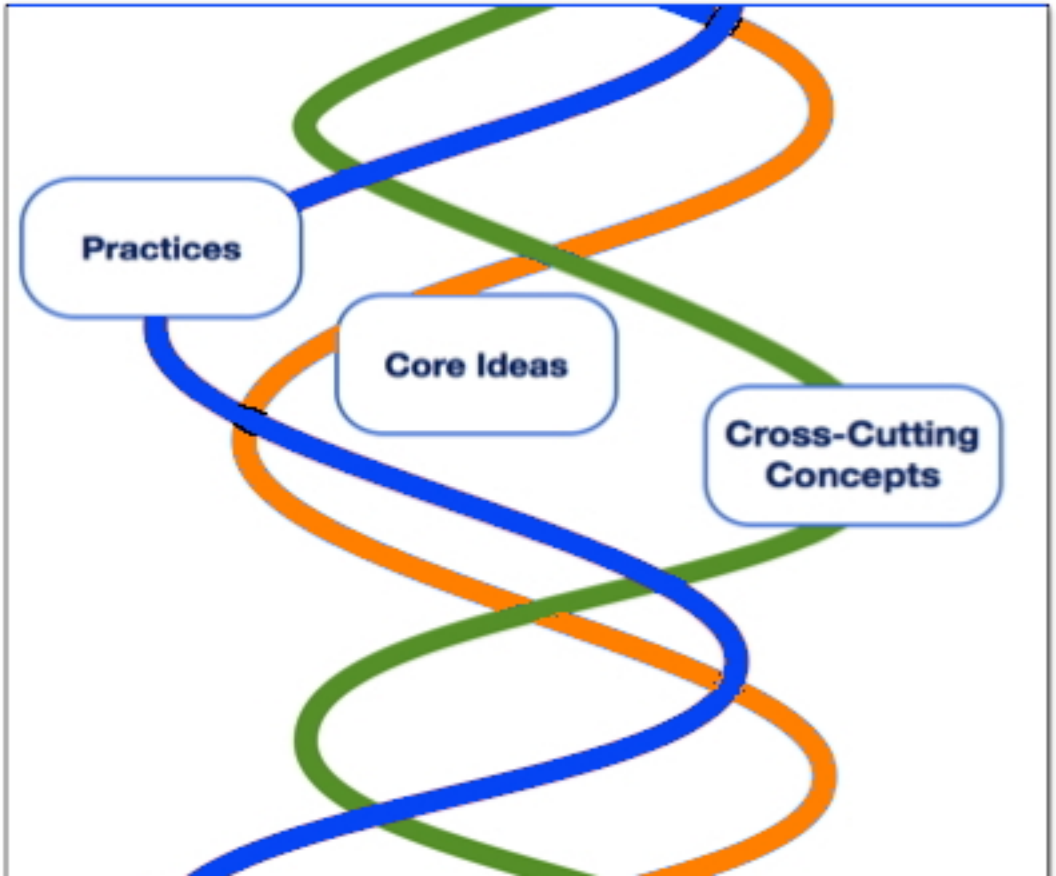
STANDARDS

For States, By States

Innovations in the NGSS

Innovations in the NGSS

1. Three-Dimensional Learning



Science and Engineering Practices

1. Asking questions and defining problems
2. Developing and using models
3. Planning and carrying out investigations
4. Analyzing and interpreting data
5. Using mathematics, information and computer technology, and computational thinking
6. Constructing explanations and designing solutions
7. Engaging in argument from evidence
8. Obtaining, evaluating, and communicating information

Innovations in the NGSS

1. Three-Dimensional Learning
2. Students Engaging in Phenomena and Designed Solutions

Innovations in the NGSS

1. Three-Dimensional Learning
2. Students Engaging in Phenomena and Designed Solutions
3. Engineering Design and Nature of Science are integrated into science

Innovations in the NGSS

1. Three-Dimensional Learning
2. Students Engaging in Phenomena and Designed Solutions
3. Engineering Design and Nature of Science are integrated into science
4. All three dimensions build coherent learning progressions

Innovations in the NGSS

1. Three-Dimensional Learning
2. Students Engaging in Phenomena and Designed Solutions
3. Engineering Design and Nature of Science are integrated into science
4. All three dimensions build coherent learning progressions
5. Science is connected to math and literacy

Science and Engineering Practices

1. Asking questions and defining problems
2. Developing and using models
3. Planning and carrying out investigations
4. Analyzing and interpreting data
5. Using mathematics and computational thinking
6. Constructing explanations and designing solutions
7. Engaging in argument from evidence
8. Obtaining, evaluating, and communicating information

Computational Thinking in the NGSS PEs

HS-PS3-1.	Create a computational model to calculate the change in the energy of one component in a system when the change in energy of the other component(s) and energy flows in and out of the system are known.
HS-LS2-1.	Use mathematical and/or computational representations to support explanations of factors that affect carrying capacity of ecosystems at different scales.
HS-LS4-6.	Create or revise a simulation to test a solution to mitigate adverse impacts of human activity on biodiversity.*
HS-ESS1-4.	Use mathematical or computational representations to predict the motion of orbiting objects in the solar system.
HS-ESS3-3.	Create a computational simulation to illustrate the relationships among management of natural resources, the sustainability of human populations, and biodiversity.
HS-ESS3-6.	Use a computational representation to illustrate the relationships among Earth systems and how those relationships are being modified due to human activity.
HS-ETS1-4.	Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.

How do we build throughout K-12 Instruction to get students there?

K–2 Condensed Practices	3–5 Condensed Practices	6–8 Condensed Practices	9–12 Condensed Practices
Mathematical and computational thinking in K–2 builds on prior experience and progresses to recognizing that mathematics can be used to describe the natural and designed world(s).	Mathematical and computational thinking in 3–5 builds on K–2 experiences and progresses to extending quantitative measurements to a variety of physical properties and using computation and mathematics to analyze data and compare alternative design solutions.	Mathematical and computational thinking in 6–8 builds on K–5 experiences and progresses to identifying patterns in large data sets and using mathematical concepts to support explanations and arguments.	Mathematical and computational thinking in 9-12 builds on K-8 and experiences and progresses to using algebraic thinking and analysis, a range of linear and nonlinear functions including trigonometric functions, exponentials and logarithms, and computational tools for statistical analysis to analyze, represent, and model data. Simple computational simulations are created and used based on mathematical models of basic assumptions.

From NGSS Appendix F

How do we build throughout K-12 Instruction to get students there?

K–2 Condensed Practices	3–5 Condensed Practices	6–8 Condensed Practices	9–12 Condensed Practices
• Decide when to use qualitative vs. quantitative data.	• Decide if qualitative or quantitative data are best to determine whether a proposed object or tool meets criteria for success.		
• Use counting and numbers to identify and describe patterns in the natural and designed world(s).	• Organize simple data sets to reveal patterns that suggest relationships.	• Use digital tools (e.g., computers) to analyze very large data sets for patterns and trends.	• Create and/or revise a computational model or simulation of a phenomenon, designed device, process, or system.
• Describe, measure, and/or compare quantitative attributes of different objects and display the data using simple graphs.	• Describe, measure, estimate, and/or graph quantities such as area, volume, weight, and time to address scientific and engineering questions and problems.	• Use mathematical representations to describe and/or support scientific conclusions and design solutions.	• Use mathematical, computational, and/or algorithmic representations of phenomena or design solutions to describe and/or support claims and/or explanations.

How do we build throughout K-12 Instruction to get students there?

K–2 Condensed Practices	3–5 Condensed Practices	6–8 Condensed Practices	9–12 Condensed Practices
• Use quantitative data to compare two alternative solutions to a problem.	• Create and/or use graphs and/or charts generated from simple algorithms to compare alternative solutions to an engineering problem.	• Create algorithms (a series of ordered steps) to solve a problem. • Apply mathematical concepts and/or processes (such as ratio, rate, percent, basic operations, and simple algebra) to scientific and engineering questions and problems. • Use digital tools and/or mathematical concepts and arguments to test and compare proposed solutions to an engineering design problem.	• Apply techniques of algebra and functions to represent and solve scientific and engineering problems. • Use simple limit cases to test mathematical expressions, computer programs, algorithms, or simulations of a process or system to see if a model "makes sense" by comparing the outcomes with what is known about the real world. • Apply ratios, rates, percentages, and unit conversions in the context of complicated measurement problems involving quantities with derived or compound units (such as mg/mL, kg/m3 , acre-feet, etc.).

Examples of Computational Modeling in the NGSS PEs

K-ESS3-1.	Use a model to represent the relationship between the needs of different plants and animals (including humans) and the places they live.
2-ESS2-2.	Develop a model to represent the shapes and kinds of land and bodies of water in an area.
4-PS4-1.	Develop a model of waves to describe patterns in terms of amplitude and wavelength and that waves can cause objects to move.
5-LS2-1.	Develop a model to describe the movement of matter among plants, animals, decomposers, and the environment.
MS-PS1-4.	Develop a model that predicts and describes changes in particle motion, temperature, and state of a pure substance when thermal energy is added or removed.
MS-LS3-2.	Develop and use a model to describe why asexual reproduction results in offspring with identical genetic information and sexual reproduction results in offspring with genetic variation.
MS-ETS1-4.	Develop a model to generate data for iterative testing and modification of a proposed object, tool, or process such that an optimal design can be achieved.
HS-PS1-4.	Develop a model to illustrate that the release or absorption of energy from a chemical reaction system depends upon the changes in total bond energy.

Examples of other PEs for which modeling and simulation can be useful tools

MS-LS2-1. Analyze and interpret data to provide evidence for the effects of resource availability on organisms and populations of organisms in an ecosystem.

What are some other places to use Computer Science principles to build toward student proficiency in NGSS PEs?

Resources:

- Nextgenscience.org/search-performance-expectations
- NGSS app
- Hard copies of the NGSS

Other examples:

MS-LS2-4. Construct an argument supported by empirical evidence that changes to physical or biological components of an ecosystem affect populations.

MS-ESS3-5. Ask questions to clarify evidence of the factors that have caused the rise in global temperatures over the past century.

HS-PS1-5: Apply scientific principles and evidence to provide an explanation about the effects of changing the temperature or concentration of the reacting particles on the rate at which a reaction occurs;

HS-ETS1-1: Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.

Supporting the intersections between Computer Science and the NGSS.

Irene Lee, Santa Fe Institute

With support from the National Science Foundation
DRL 0639637, OCI 1057672, and CNS 1240992.

The Need

- CS is a foundational part of the practice of science and engineering
- CS contributes to advancement in every field of Science
- Computational science seen as 3rd leg of Science in addition to theoretical & experimental approaches.

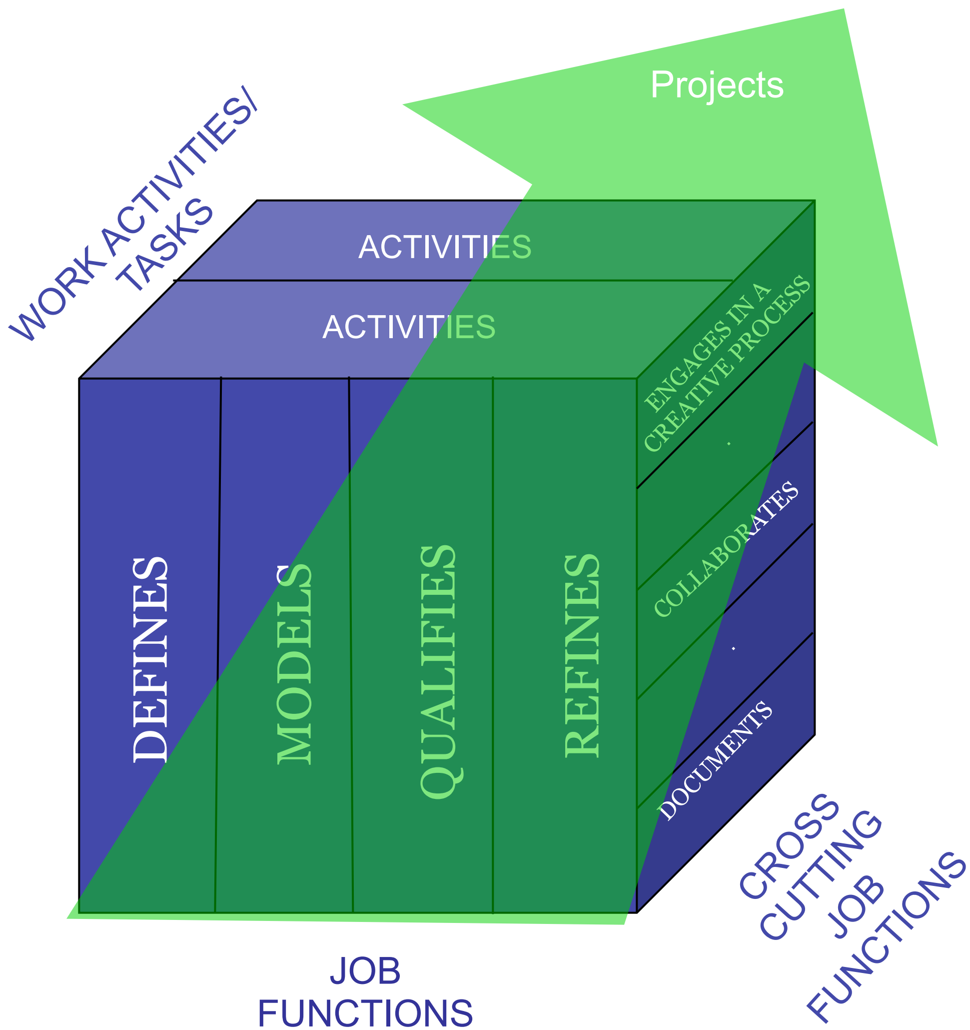
President's Information Technology Advisory Committee's Report to the President, *Computational Science: Insuring America's Competitiveness (2005).*

Computational Thinking in STEM

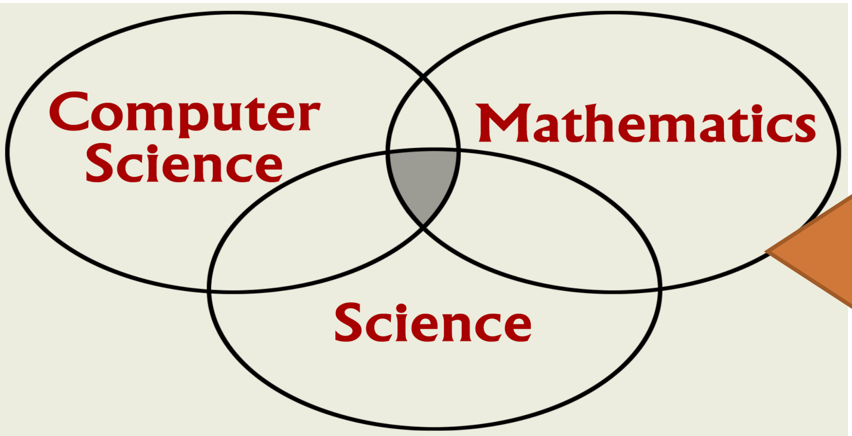
A **COMPUTATIONAL THINKING ENABLED STEM WORKER**:

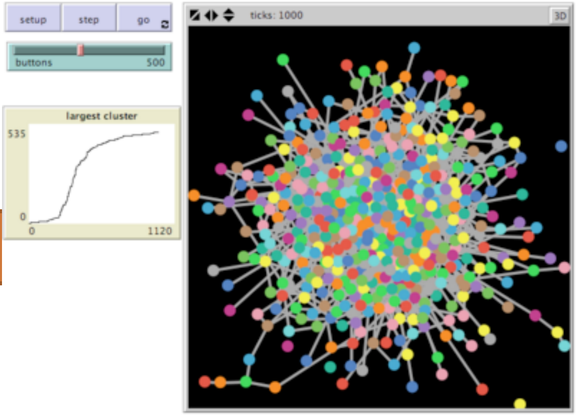
•engages in a creative process to solve problems, design products, automates systems, or improve understanding by defining, modeling, qualifying and refining systems, processes or mechanisms generally through the use of computers.

Computational thinking often occurs in collaboration with others.



Scientist's toolkit





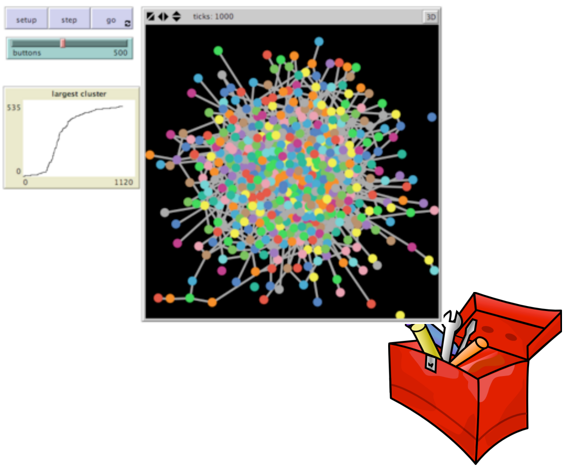
New Sub-fields:

- Computational Biology
- Computational Physics
- Computational Social Science
- Computational Chemistry

Computer Modeling and Simulation:

- Agent based modeling
- Stochastic modeling
- Monte Carlo simulation
- Systems dynamics modeling

Computational Thinking



Computer Modeling and Simulation:

- Agent based modeling
- Stochastic modeling
- Monte Carlo simulation
- Systems dynamics modeling

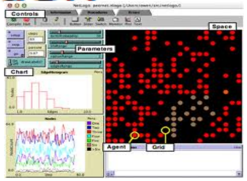
Computational Thinking:

- Abstraction
- Automation
- Analysis

Our Challenge

- to prepare computational thinking-enabled students who are able “to formulate problems so that their solutions can be represented as computational steps or algorithms to be carried out by a computer.”

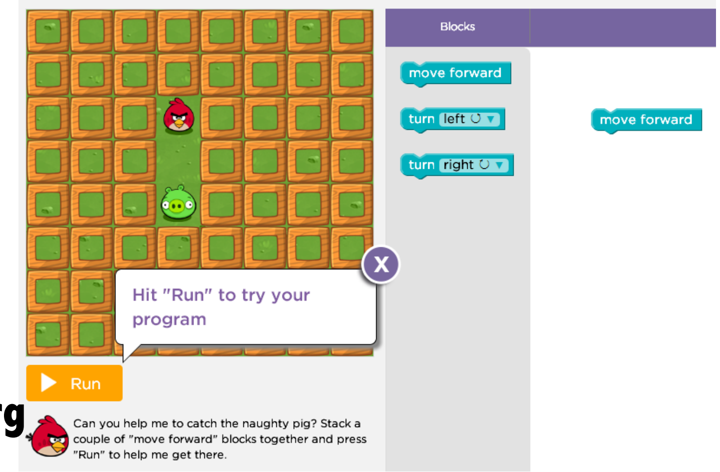
Computational Thinking in K-12

NSF EDC	Abstraction	Automation	Analysis
Modeling & Simulation 	Selecting features of real-world to incorporate in a model.	Time stepping using a model as an experimental testbed.	Were the correct abstractions made? Does the model reflect reality?
Robotics 	Design robot to react to a set of conditions.	Program checks sensors to monitor conditions.	Are there situations that were not taken into account?
Game Design & Development 	Games are abstracted into a set of scenes containing characters	Game responds to user actions.	Do the elements incorporated make the game fun to play?

Supporting Growth in CT

- Provide opportunities to learn and practice basic CS concepts and constructs

CODE
learn.code.org



Thinking about models

- Comparing and contrasting real world "participatory simulations" and the same activities within a computer model.

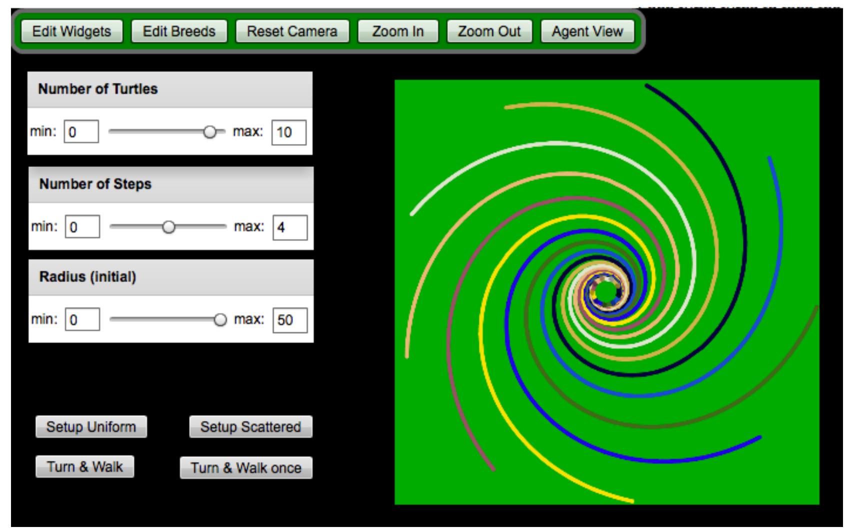
Setup in a large circle (at least 4 ft away from your neighbors)

Then, at my command, follow these two instructions:

1. Turn to the person on your left and set that direction as your heading;
2. Take 3 steps forward in that direction.

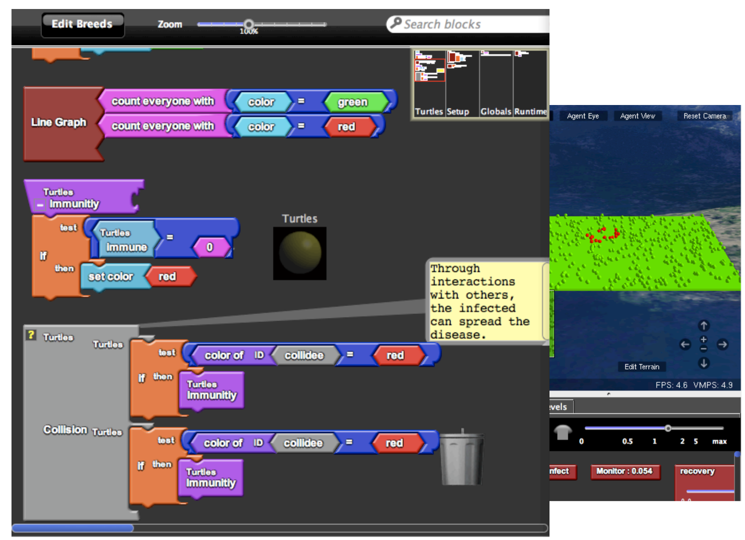
Thinking about models

- Comparing and contrasting real world “participatory simulations” and the same activities within a computer model.

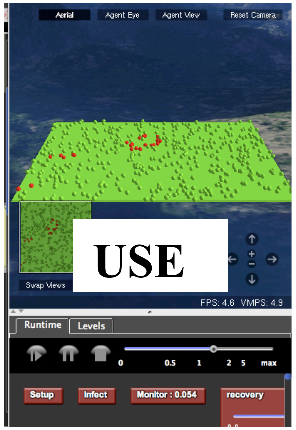


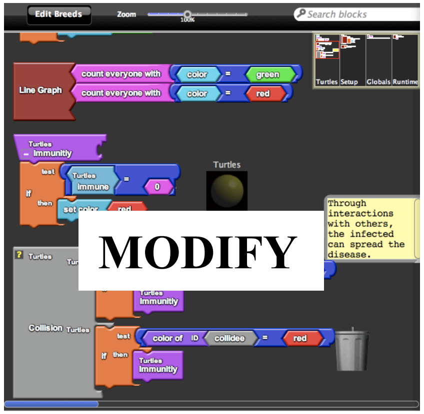
Supporting Growth in CT

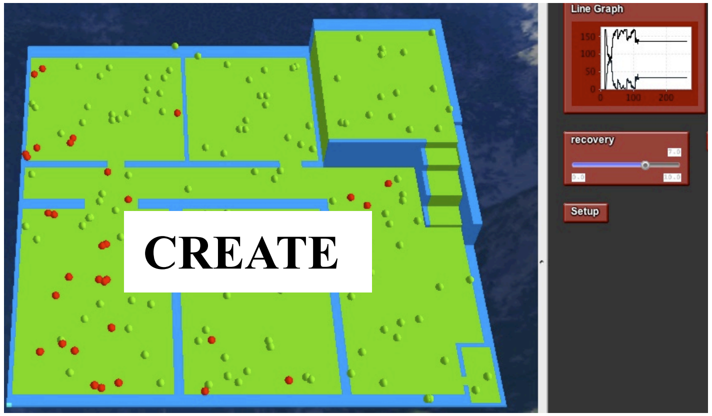
- **Rich computational environments** allow users to "look under the hood" and see how it works.
  - *Ex) Scratch for creating animations that respond to user input*
  - *Ex) StarLogo TNG for creating models of complex systems*

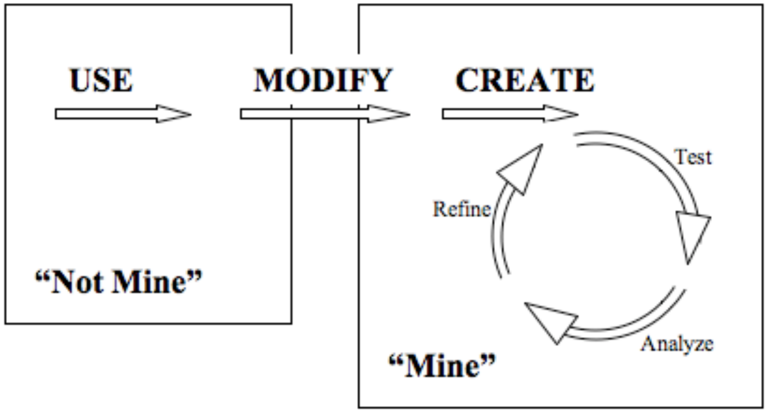


Use-Modify-Create progression



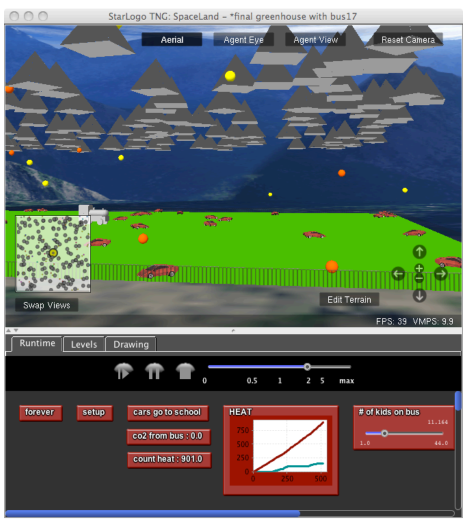


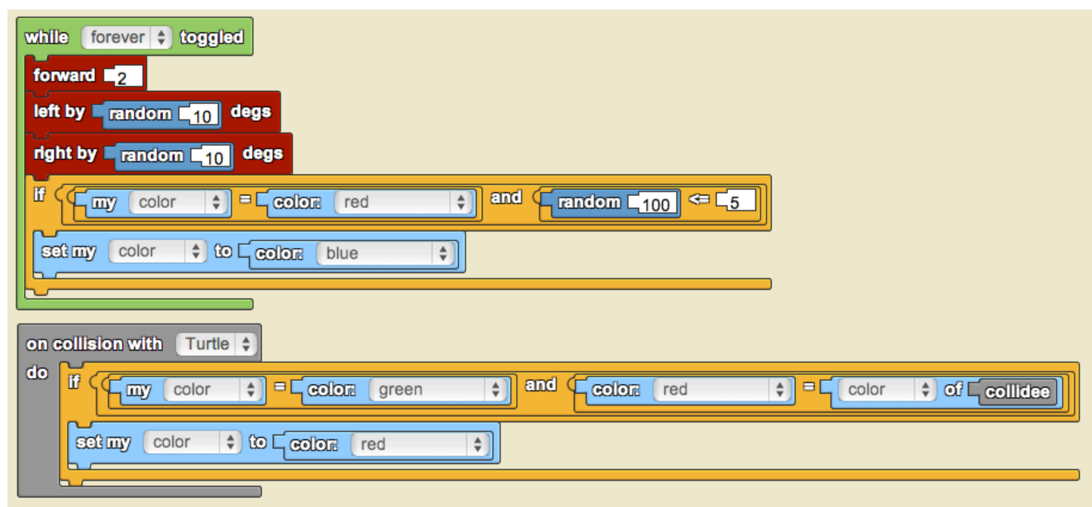




Modeling and Simulation

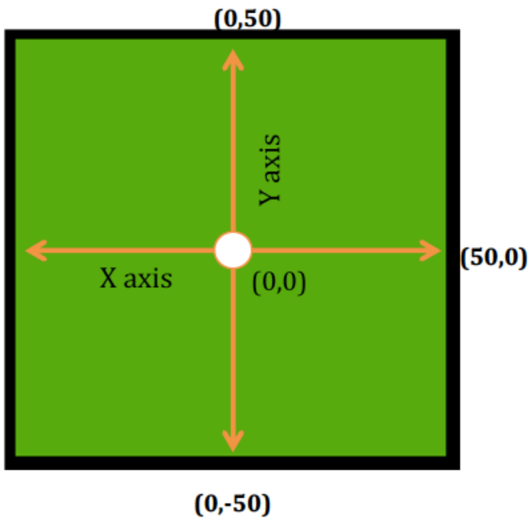
- **Agent based modeling**
  - **StarLogo TNG (*education.mit.edu/starlogo-tng*)**
  - **StarLogo Nova (slnova.org)**
  - **NetLogo (*ccl.northwestern.edu/netlogo*)**

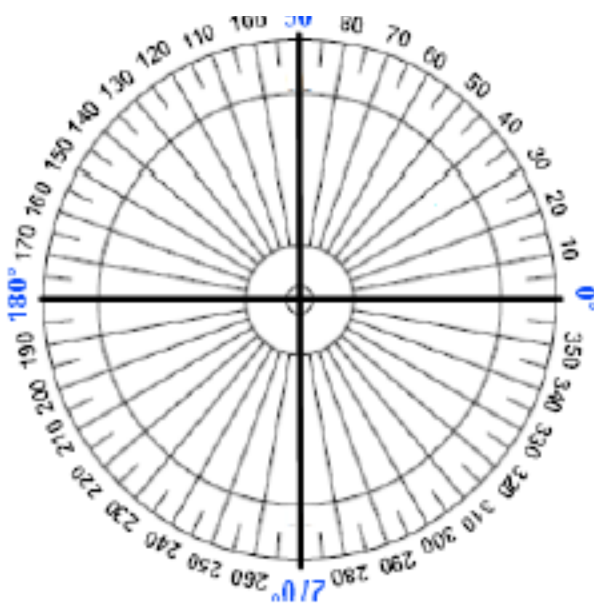




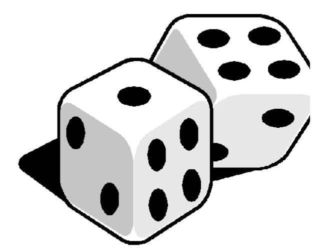
Making links to mathematics

- Geometry
- Logic
- Probability and Statistics





		White Die					
		1	2	3	4	5	6
Red Die	1	(1,1)	(2,1)	(3,1)	(4,1)	(5,1)	(6,1)
	2	(1,2)	(2,2)	(3,2)	(4,2)	(5,2)	(6,2)
	3	(1,3)	(2,3)	(3,3)	(4,3)	(5,3)	(6,3)
	4	(1,4)	(2,4)	(3,4)	(4,4)	(5,4)	(6,4)
	5	(1,5)	(2,5)	(3,5)	(4,5)	(5,5)	(6,5)
	6	(1,6)	(2,6)	(3,6)	(4,6)	(5,6)	(6,6)



Comparing mathematical models to agent based models

Basic SIR Model

$$\frac{dS}{dt} = -\beta SI \qquad \frac{dI}{dt} = \beta SI - \gamma I \qquad \frac{dR}{dt} = \gamma I$$

The influence of age: age-structured models

[edit]

Maybe "the most specific parameter of biological system is the age" (M. Iannelli), and, especially for some infectious diseases, it has a deep influence on the dynamics of its spreading in a population. Many of the parameters we have seen may depend on age, and especially the contact rate, which summarizes the 'infectious effectiveness' of contacts between susceptible and infectious subjects. This effectiveness has, thus, to take into account both the age of the infectious and the age of the susceptible. Epidemic models modeling the age structure of a population are very complex. In fact, they are infinite dimensional model since we have to deal with density through the ages of the epidemic classes $s(t,a), i(t,a), r(t,a)$ (to limit ourselves to the susceptible-infectious-removed scheme) such that:

$$S(t) = \int_0^{a_M} s(t,a)da$$

$$I(t) = \int_0^{a_M} r(t,a)da$$

$$R(t) = \int_0^{a_M} r(t,a)da$$

(where $a_M \leq +\infty$ is the maximum admissible age)and their dynamics is not described, as one might think, by "simple" partial differential equations, but by integro-differential equations:

$$\partial_t s(t,a) + \partial_a s(t,a) = -\mu(a)s(a,t) - s(a,t)\int_0^{a_M} k(a,a_1;t)i(a_1,t)da_1$$

$$\partial_t i(t,a) + \partial_a i(t,a) = s(a,t)\int_0^{a_M} k(a,a_1;t)i(a_1,t)da_1 - \mu(a)i(a,t) - \nu(a)i(a,t)$$

$$\partial_t r(t,a) + \partial_a r(t,a) = -\mu(a)r(a,t) + \nu(a)i(a,t)$$

where:

$$F(a,t,i(.,.)) = \int_0^{a_M} k(a,a_1;t)i(a_1,t)da_1$$

is the force of infection, which, of course, will depend, though the contact kernel $k(a,a_1;t)$ on the interactions between the ages.

Complexity is added by the initial conditions for newborns (i.e. for a=0), that are straightforward for infectious and removed:

$i(t,0) = r(t,0) = 0$

but that are nonlocal for the density of susceptible newborns:

$$s(t,0) = \int_0^{a_M} \left(\varphi_s(a)s(a,t) + \varphi_i(a)i(a,t) + \varphi_r(a)r(a,t)\right) da$$

where $\varphi_j(a), j = s, i, r$ are the fertilities of the adults.

Moreover, defining now the density of the total population $n(t,a) = s(t,a) + i(t,a) + r(t,a)$ one obtains:

$$\partial_t n(t,a) + \partial_a n(t,a) = -\mu(a)n(a,t)$$

In the simplest case of equal fertilities in the three epidemic classes, we have that in oder to have demographic equilibrium the following necessary and sufficient condition linking the fertility $\varphi(.)$ with the mortality $\mu(a)$ must hold:

$$1 = \int_0^{a_M} \varphi(a) Exp\left(-\int_0^a \mu(q)dq\right) da$$

and the demographic equilibrium is

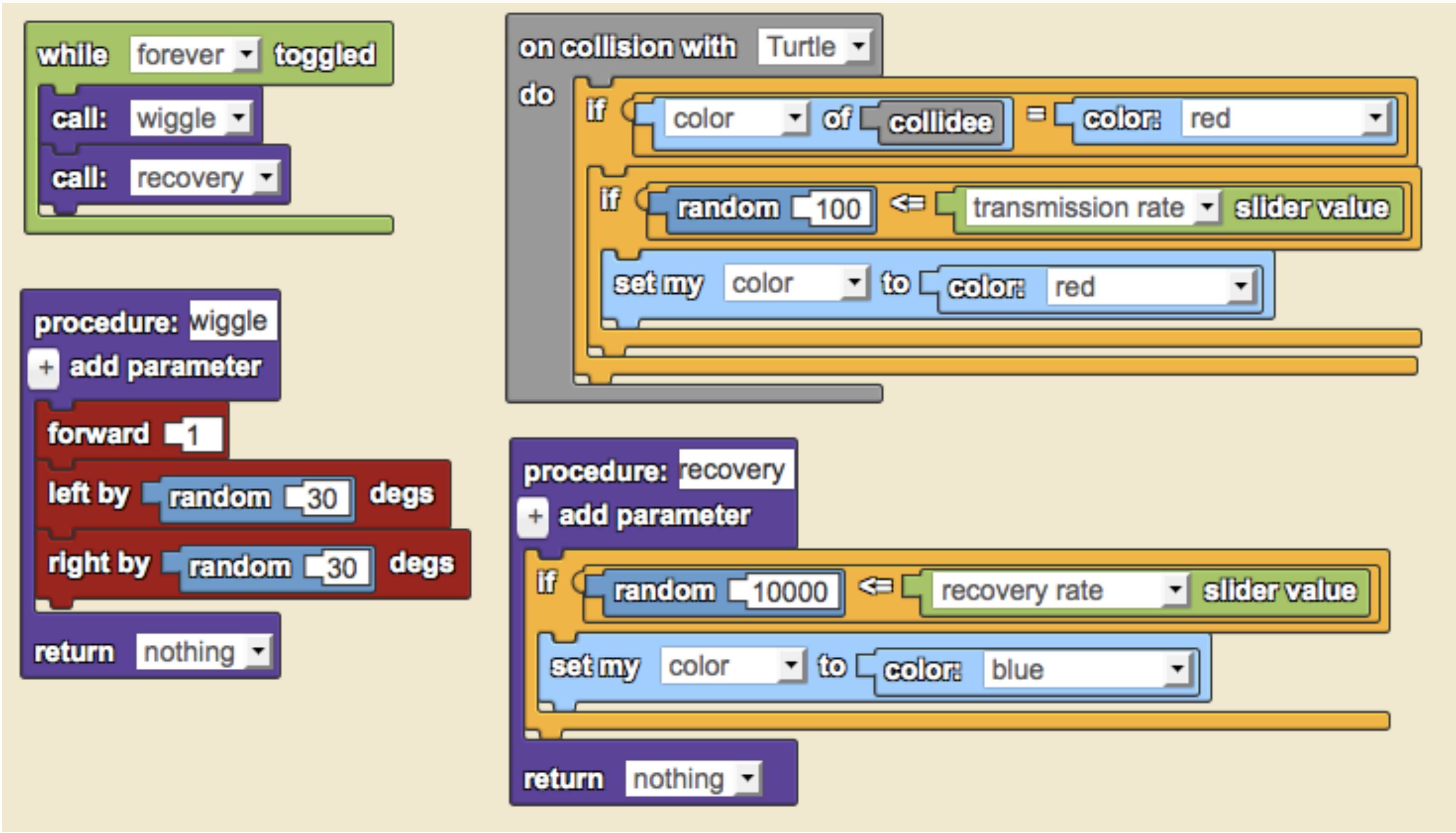
$$n^*(a) = CExp\left(-\int_0^a \mu(q)dq\right),$$

automatically ensuring the existence of the disease-free solution:

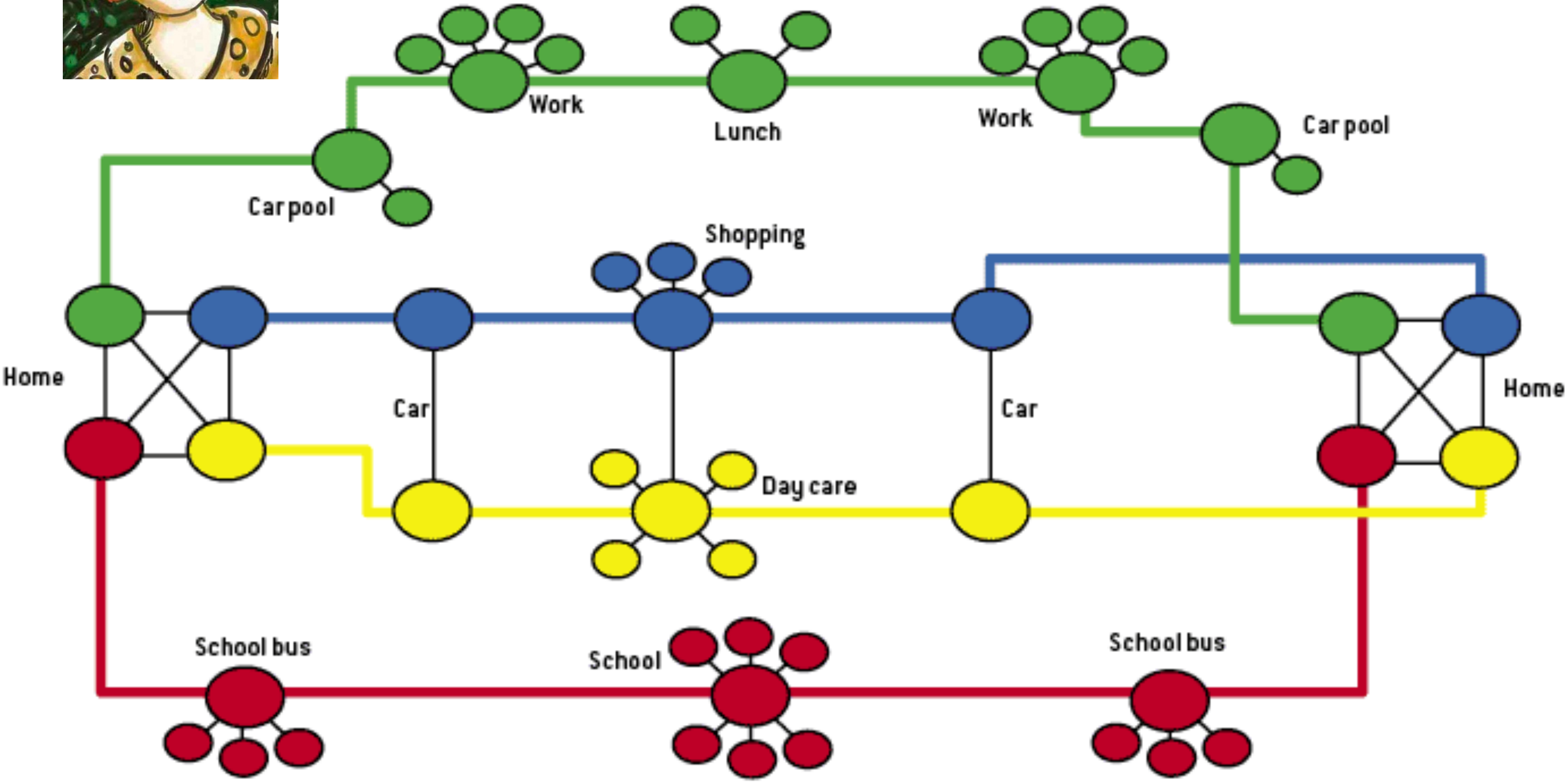
$$DFS(a) = (n^*(a), 0, 0).$$

You can make these systems of equations REALLY complicated.

Basic SIR Model



Agent Based Model



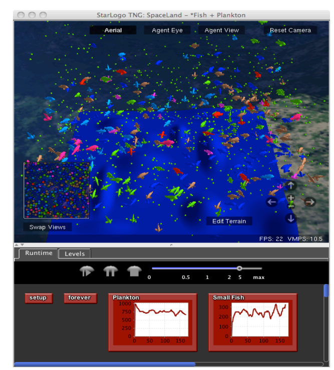
EpiSims: created at Los Alamos

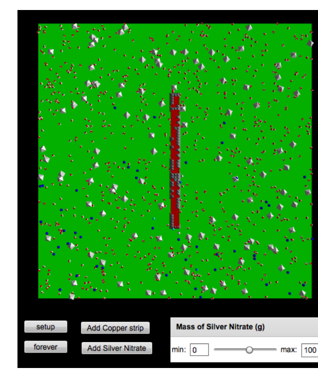
Challenges encountered

- Integration into existing curricula during the regular school day is difficult.
- Incorporating CS takes too much time. It takes extra time for students to create, innovate, communicate, and collaborate on projects driven by their own interests and questions.
- Getting buy-in of administrators.
- Professional development for teachers.

Integrating "CS in Science"

- *4 curricular modules by Project GUTS for Code.org*
  - *Introduction to Modeling and Simulation*
  - *Earth Science – Water resources*
  - *Life Science – Ecosystems as complex systems*
  - *Physical Science – Chemical reactions*
- *Each is a 5 hour "replacement" unit that follows the Use-Modify-Create trajectory*





Barriers to overcome:

Time constraints https://www.youtube.com/watch?v=bfgsdyxEzxM
Is what students are learning tested? https://www.youtube.com/watch?v=j2wMZ-3hARo
Getting district buy-in District buy in https://www.youtube.com/watch?v=2icrW3QRkdM

Opportunities afforded by CS in Science

- Student plays role of theorist and experimentalist
- Authentic experience of science, asking questions and making discoveries

Ex) Ecosystem dynamics

Q: How many different patterns of population growth and death can be found? What are they?

Q: What is the impact of decreasing the maximum number of plankton in the ecosystem?

Q: How can I achieve a long lasting ecosystem?

Q: What is the impact of adding a top predator?

Opportunities afforded by CS in Science

MS-LS2-1. Analyze and interpret data to provide evidence for the effects of resource availability on organisms and populations of organisms in an ecosystem.

MS-LS2-4. Construct an argument supported by empirical evidence that changes to physical or biological components of an ecosystem affect populations.

MS-LS2-2. Construct an explanation that predicts patterns of interactions among organisms across multiple ecosystems.

Opportunities afforded by CS in Science

- Supports argumentation from evidence.
- Using models as test beds generates evidence.

Ex) Argument from evidence: is variation in a population necessary before selection can act?

HS-LS4-3: Apply concepts of statistics and probability to support explanations that organisms with an advantageous heritable trait tend to increase in proportion to organisms lacking this trait;

HS-LS4-4: Construct an explanation based on evidence for how natural selection leads to adaptation of populations

Opportunities afforded by CS in Science

- Students who construct their own simulations will display a clear grasp of scientific concepts expected in the NGSS.

Ex) Chemical reactions

- the conditions under which they occur,
- the evidence of a chemical reaction,
- limiting reactants versus reactants in excess, and
- when chemical reactions stop.

Demonstration

DISCUSSION

- Q: Can today's STEM teachers become tomorrow's Computational Thinking enabled teachers who understand and can facilitate students in these practices?
- Q: At the end, say "now that you've heard about CT and how it can be integrated in Science through M&S, what misconceptions do you think your administrator or other teachers have about CT?"
- Q: Do these misconceptions create barriers to implementation, and if so, how can they be addressed? diffused? knocked down.
- Think of your curriculum - are there units or activities that you do now that you think could be strengthened through integrating computational thinking, modeling and simulation.

Thank you!

Josh Caldwell, Code.org	josh@code.org
Jennifer Childress, Achieve	JChildress@achieve.org
Irene Lee, MIT	ialee@mit.edu

Websites:

- Code.org/curriculum/science
- Nextgenscience.org
- Projectguts.org