

## **Curriculum Map: Discovering Computer Science 8**

### **Cochranton Junior and Senior High School**

#### **Business Department**

**Course Description:** Computer Science and computational problem solving are fundamental skills for engaging the 21st-century marketplace of ideas and economies. We believe that all students should have the opportunity to learn these skills as they will use them in whatever career they are likely to enter. **CS0** is a light version of the Carnegie Science course CS1 designed to engage and excite future programming students. It includes 16 lessons and is intended for middle school rotation settings.

This free CS0 curriculum is designed for students in 8th grade with algebra readiness skills. No prior programming experience is required. It is inspired by a highly successful Intro Computing course (15-112, Fundamentals of Programming and Computer Science) that has been taught at Carnegie Mellon University for the past 10+ years. It is predicated on the notion that learning about programming and computer science should be fun and engaging. This requires interesting problems to solve, as computational problem-solving is the core of computer science.

- Creativity
- Innovation
- Critical Thinking
- Problem Solving
- Communication
- Teamwork
- Collaboration

## **Standards Recognized through the Course**

### **The state of Pennsylvania has adopted CSTA Standards for computer science.**

#### **CSTA Standards:**

##### **Algorithms & Programming**

###### **Program Development & Creating**

2-AP-11 Create clearly named variables that represent different data types and perform operations on their values

2-AP-19 Document programs in order to make them easier to follow, test, and debug

3A-AP-139-10 Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.

3A-AP-169-10 Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.

3A-AP-199-10 Systematically design and develop programs for broad audiences by incorporating feedback from users.

###### **Computational Problems**

2-AP-13 Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.

3A-AP-179-10 Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.

###### **Testing**

2-AP-17 Systematically test and refine programs using a range of test cases.

3A-AP-219-10 Evaluate and refine computational artifacts to make them more usable and accessible.

###### **Collaborating & Communicating**

2-AP-15 Seek and incorporate feedback from team members and users to refine a solution that meets user needs.

3A-AP-229-10 Design and develop computational artifacts working in team roles using collaborative tools.

3A-AP-239-10 Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs.

## **ISTE Standards for Students:**

### **1. Empowered Learner**

Students leverage technology to take an active role in choosing, achieving and demonstrating competency in their learning goals, informed by the learning sciences.

Students:

- A. articulate and set personal learning goals, develop strategies leveraging technology to achieve them and reflect on the learning process itself to improve learning outcomes.
- B. build networks and customize their learning environments in ways that support the learning process.
- C. use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.

### **4. Innovative Designer**

Students use a variety of technologies within a design process to identify and solve problems by creating new, useful or imaginative solutions.

Students:

- A. know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts or solving authentic problems.
- B. develop, test and refine prototypes as part of a cyclical design process.
- C. exhibit a tolerance for ambiguity, perseverance and the capacity to work with open-ended problems.

### **5. Computational Thinker**

Students develop and employ strategies for understanding and solving problems in ways that leverage the power of technological methods to develop and test solutions.

Students:

- A. break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving.

### **6. Creative Communicator**

Students communicate clearly and express themselves creatively for a variety of purposes using the platforms, tools, styles, formats and digital media appropriate to their goals.

Students:

- A. create original works or responsibly repurpose or remix digital resources into new creations.

- B. communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations, models or simulations.
- C. publish or present content that customizes the message and medium for their intended audiences.

**Unit 1 Title:**                      **Drawing with Shapes**

**Suggested time frame:**              3 weeks

**Standards Addressed:**

- **CSTA Standards:**

- **2-AP-19**
- **3A-AP-139-10**
- **3A-AP-199-10**
- **2-AP-13**
- **2-AP-17**
- **2-AP-15**

- **ISTE Standards for Students:**

- **1. Empowered Learner**
- **4. Innovative Designer**
- **5. Computational Thinker**
- **6. Creative Communicator**

**Essential Questions:**

1. What are students' individual interests and how can computing enhance those interests?
2. What is programming?
3. How can students use objects to draw and create illustrations through code?
4. How is programming used to create, compare, represent, and model shapes?

Competency	Vocabulary	Strategy	Resources
<ul style="list-style-type: none"> <li>• Define the Canvas</li> <li>• Draw Objects using the Python programming language</li> <li>• Create a program using the Python programming language</li> <li>• Use Algorithms in a program</li> <li>• Draw a Scene on a Canvas by manipulating Drawing objects, parameters, and properties</li> <li>• Test and Debug programs</li> </ul>	<ul style="list-style-type: none"> <li>• Python</li> <li>• Canvas</li> <li>• Coordinates</li> <li>• The Inspector</li> <li>• Errors</li> <li>• Debugging</li> <li>• Objects</li> <li>• Gradients</li> <li>• Properties</li> <li>• Parameters</li> </ul>	<p>Class discussions and modeling will be used to introduce new content. CSU site will be utilized to complete task “checkpoints” for current understanding of individual new concepts. Exercises will be completed for formative assessment. Creative tasks will be utilized for cumulative assessments</p>	<p>Internet access Laptop / Thin Client / Desktop Carnegie Science University – online course (free district resource) <a href="https://academy.cs.cmu.edu/course/cs0">https://academy.cs.cmu.edu/course/cs0</a> Notebook or Folder Planning guides</p>

**Unit 2 Title:** Basic Animations

**Suggested time frame:** 2 weeks

• **CSTA Standards:**

- 2-AP-11
- 2-AP-19
- 3A-AP-139-10
- 3A-AP-169-10
- 3A-AP-199-10
- 2-AP-13
- 2-AP-17

▪ **ISTE Standards for Students:**

- Empowered Learner
- Innovative Designer
- Computational Thinker
- Creative Communicator

**Essential Questions:**

1. What is object-oriented Programming
2. What is an event?
3. How can we use code to make objects move
4. What is an animation
5. What is a variable
6. How do you use a variable as a counter
7. What is abstraction?

Competency	Vocabulary	Strategy	Resources
<ul style="list-style-type: none"> <li>• Define Object Oriented programming and its purpose</li> <li>• Define animation</li> <li>• Define Abstraction</li> <li>• Create Variables</li> <li>• Use Variables in Code to maintain high levels of versatility</li> <li>• Code mouse events to create a more functional and interactive user program</li> </ul>	<ul style="list-style-type: none"> <li>• Object Oriented Programming</li> <li>• Abstraction</li> <li>• Global Variables</li> <li>• Mouse Events</li> <li>• onMousePress</li> <li>• onMouseRelease</li> <li>• onStep</li> </ul>	<p>Class discussions and modeling will be used to introduce new content. CSU site will be utilized to complete task “checkpoints” for current understanding of individual new concepts. Exercises will be completed for formative assessment. Creative tasks will be utilized for cumulative assessments</p>	<p>Internet access Laptop / Thin Client / Desktop Carnegie Science University – online course (free district resource) <a href="https://academy.cs.cmu.edu/course/cs0">https://academy.cs.cmu.edu/course/cs0</a> Notebook or Folder Planning guides</p>



**Unit 3 Title:** Conditionals and Animations

**Suggested time frame:** 2 weeks

• **CSTA Standards:**

- 2-AP-11
- 2-AP-19
- 3A-AP-139-10
- 3A-AP-169-10
- 3A-AP-199-10
- 2-AP-13
- 2-AP-17

▪ **ISTE Standards for Students:**

- **Empowered Learner**
- **Innovative Designer**
- **Computational Thinker**
- **Creative Communicator**

**Essential Questions:**

1. How can students use visualizations to create interactive games?
2. How can students use animations to accomplish game-like tendencies?
3. How do you make decisions in a Python Program?

Competency	Vocabulary	Strategy	Resources
<ul style="list-style-type: none"><li>• Create more complex animations</li><li>• Use animations to mimic typical video-games</li><li>• Collaborate on program with a partner using Pair Programming</li><li>• Use more complex properties to create advanced animation visuals</li><li>• Use the hitShape method to detect collisions of objects</li></ul>	<ul style="list-style-type: none"><li>• Conditionals</li><li>• If key word</li><li>• onKeyPress</li><li>• key Parameter</li><li>• hitShape</li><li>• onStep</li><li>• timers</li><li>• toggle</li></ul>	Class discussions and modeling will be used to introduce new content. CSU site will be utilized to complete task “checkpoints” for current understanding of individual new concepts. Exercises will be completed for formative assessment. Creative tasks will be utilized for cumulative assessments	Internet access Laptop / Thin Client / Desktop Carnegie Science University – online course (free district resource) <a href="https://academy.cs.cmu.edu/course/cs0">https://academy.cs.cmu.edu/course/cs0</a> Notebook or Folder Planning guides Portfolio

**Unit 4 Title:** Games

**Suggested time frame:** 2 weeks

• **CSTA Standards:**

- 2-AP-11
- 2-AP-19
- 3A-AP-139-10
- 3A-AP-169-10
- 3A-AP-199-10
- 2-AP-13
- 2-AP-17

• **3A-AP-239-10**

▪ **ISTE Standards for Students:**

- **Empowered Learner**
- **Innovative Designer**
- **Computational Thinker**
- **Creative Communicator**

**Essential Questions:**

1. What is an iteration Structure?
2. How does the for loop act as an iteration structure?
3. What is the body of a loop?
4. How do loops increase the function of a program?
5. How can you create more efficient code using abstraction via local variables?

Competency	Vocabulary	Strategy	Resources
<ul style="list-style-type: none"><li>• Use for loops to create complex iteration structures to handle repetitive code events</li><li>• Use for loops to create a functional computerized game in Python</li><li>• Collaborate on program with a partner using Pair Programming</li><li>• Create a computational Artifact that showcase your programming portfolio</li></ul>	<ul style="list-style-type: none"><li>• Iteration structure</li><li>• For loop</li><li>• Body</li><li>• Range</li><li>• Local variable</li></ul>	Class discussions and modeling will be used to introduce new content. CSU site will be utilized to complete task “checkpoints” for current understanding of individual new concepts. Exercises will be completed for formative assessment. Creative tasks will be utilized for cumulative assessments	Internet access Laptop / Thin Client / Desktop Carnegie Science University – online course (free district resource) <a href="https://academy.cs.cmu.edu/course/cs0">https://academy.cs.cmu.edu/course/cs0</a> Notebook or Folder Planning guides Portfolio

- |   |  |  |  |
|---|--|--|--|
| <ul style="list-style-type: none"><li>• Share programming artifact portfolio with peers</li></ul> |  |  |  |
|---|--|--|--|