

# STRANDS AND STANDARDS

## CREATIVE CODING



### Course Description

*Creative Coding through Games and Apps* is a first-semester course for introduction to programming for the early secondary grades. The course is designed to attract and reach a broad and diverse range of students, including those who may have never considered programming. Students learn how to code by working in a real software development environment to design, program and publish mobile apps and games. Learning to code by creating real products, students discover how to make amazing things and have an impact on their world.

<b>Intended Grade Level</b>	6-8
Units of Credit	0.5
Core Code	35.02.00.00.003
Concurrent Enrollment Core Code	N/A
Prerequisite	N/A
Skill Certification Test Number	N/A
Test Weight	0.0
<b>License Type</b>	CTE and/or Secondary Education 6-12
<b>Required Endorsement(s)</b>	
Endorsement 1	No endorsement required

## STRAND 1

**Students demonstrate knowledge of basic structure, elements, and logic of computer games.**

### Standard 1

Understand the concept of a computer program in relation to scripts, libraries, related data and software.

### Standard 2

Identify computer game genres. Describe characteristics and elements of successful games. Describe multiple ways that games are used to educate and inform.

### Standard 3

Analyze complete computer game flow by explaining the rules, goals, libraries and structure of a computer game.

### Standard 4

Analyze and predict program code.

### Standard 5

Identify characteristics of problems that are hard for computers to solve, then decompose a familiar problem into simple steps by illustrating a process with precision and clarity.

### Standard 6

Interpret the flow and execution of algorithms and predict their outcomes, then create algorithms using pseudo code.

### Standard 7

Apply gaming coordinate system knowledge by identifying code used to create and position sprites on a game board, then write code to position sprites.

## STRAND 2

**Students explore randomization in games.**

### Standard 1

Analyze a game script to discover how random numbers are used in a game.

### Standard 2

Describe what is meant by a random number within a range

### Standard 3

Discover how random numbers are used in a game, then write code to use a random range function to generate random numbers

## STRAND 3

**Students control game objects.**

### Standard 1

Identify, explain, and define an object, then write the code for an object in a computer program.

### Standard 2

Identify, explain, and define a function, then write the code for a function in a computer program.

### Standard 3

Identify, explain, and define a parameter, then write the code for a parameter in a function.

### Standard 4

Identify, explain, and define a variable, then write the code to declare and use a variable in a computer program.

### Standard 5

Create variables that represent different types of data and manipulate their values.

### Standard 6

Give examples of user input, then utilize input and variables to calculate new information.

### Standard 7

Identify, explain and define a custom function, then write the code for a custom function in a computer program.

## STRAND 4

### Students control game flow.

#### Standard 1

Identify, explain and define an event, then write the code for an event in a computer program.

#### Standard 2

Identify, explain and define a conditional in computer program.

#### Standard 3

Compare and contrast an if-then statement and an if-else statement.

#### Standard 4

Write the code for conditionals/if statements in a computer program.

#### Standard 5

Identify, explain, and define Boolean logic, then write complex conditional statements using and, or, and not.

#### Standard 6

Identify and explain the concept of on every frame, then write code using on every frame code.

### Standard 7

Identify instances when conditionals and the on every frame code are needed.

### Standard 8

Identify, explain and define a loop, then write the code for a loop in a computer program.

## STRAND 5

**Students add enhancements to games and make them their own.**

### Standard 1

Describe enhancements that make games more appealing or interesting by identifying elements such as creative openings, background music, animations, x and y repositioning, angles, bounce on all sides, gravity/acceleration etc. then write code using several enhancements.

### Standard 2

Demonstrate how a series of individual frames can create animated images by creating a paper prototype of a sprite sheet.

### Standard 3

Create a sprite sheet using an online image editor, then use a sprite sheet in game code. Change the parameters of the set frame grid to work with different sprite sheets.

### Standard 4

Identify nonexecutable lines of code in a script such as developer comments, explanations, or instructions; then add several nonexecutable lines of code to a script to organize, explain, or instruct

## STRAND 6

**Students explore basic App design and creation.**

### Standard 1

Distinguish between apps and games.

### Standard 2

Apply basic app design concepts such as creating, loading and arranging pages as well as arranging elements on a page.

### Standard 3

Pass data to an App page using a parameter.

### Standard 4

Understand how tables are used in the context of App pages.

## STRAND 7

**Students participate in independent project development and collaboration.**

### Standard 1

Develop programs, both independently and collaboratively, that include sequences with nested loops and multiple branches using block-based and/or text-based programming languages.

### Standard 2

Describe the process for giving, receiving, revising and incorporating constructive feedback.

### Standard 3

Contribute to a collaboratively created presentation.

### Standard 4

Use an iterative design process that applies concepts learned to a project; e.g.

- Analyze a problem for a specific audience in terms of how an app or game can contribute to the solution
- Develop a schedule, track project progress, and document with journals
- Generate ideas
- Gather input
- Create storyboard
- Construct algorithm(s)
- Build program
- Articulate instructions for game play
- Test program
- Present program
- Solicit and integrate peer feedback
- Refine program
- Evaluate and reflect on learning.

### Standard 5

Provide proper attribution when code is borrowed or built upon.

### Workplace Skills

Communication, Problem Solving, Teamwork, Critical Thinking