GEORGIA | PEACH STATE PATHWAYS

Career, Technical, & Agricultural Education

BUSINESS & COMPUTER SCIENCE

PATHWAY: Computing

COURSE: Computing in the Modern World

UNIT: 6-Programming and Problem Solving (Option 2)

INTRODUCTION

Annotation: This unit attempts to introduce problem-solving, algorithm development, and programming concepts (including object-oriented programming concepts) for Computing in the Modern World. Methods used will include lecture, demonstration, individual work, group work, and project-based learning. Students will use many different pieces of technology, including productivity software, programming tools, the Internet, audio and video files, and modeling tools.

Grade(s):

X	9th
X	10th
X	11th
X	12th

Time: 10 hours

Author: Jason Naile

Additional Author(s):

Students with Disabilities:

For students with disabilities, the instructor should refer to the student's IEP to be sure that the accommodations specified are being provided. Instructors should also familiarize themselves with the provisions of Behavior Intervention Plans that may be part of a student's IEP. Frequent consultation with a student's special education instructor will be beneficial in providing appropriate differentiation.

FOCUS STANDARDS

GPS Focus Standards: Please list the standard and elements covered.

BCS-CMW-16 Students will discuss examples that identify the broad interdisciplinary utility of computers and algorithmic problem solving in the modern world.

a. List the different ways computers are used.

b. Define algorithm.

c. Identify examples of algorithmic problem solving in everyday life.

BCS-CMW-17 Students will apply strategies to solve various problems.

a. Solve a variety of logic problems and identify the strategies used.

b. List strategies for solving problems.

BCS-CMW-18 Students will apply algorithmic thinking to solve problems.

a. Evaluate algorithmic definitions for various problems and identify errors and or weaknesses.

b. Correct algorithmic definitions for various problems.

c. Complete, evaluate, and adjust an algorithm for a problem.

BCS-CMW-19 Students will demonstrate an understanding of the basic steps in algorithmic problem solving.

a. Formulate a formal problem statement.

b. Explore the problem using strategies.

c. Communicate the design of an algorithm.

d. Code, test, and verify a solution.

GPS Academic Standards:

ELA10W1 The student produces writing that establishes an appropriate organizational structure, sets a context and engages the reader, maintains a coherent focus throughout, and signals closure.

ELA10W2 The student demonstrates competence in a variety of genres.

ELA10LSV1 The student participates in student-to-teacher, student-to-student, and group verbal interactions.

SCSh3 Students will identify and investigate problems scientifically.

ELA10RL5 The student understands and acquires new vocabulary and uses it correctly in reading and writing.

ELA10RC3 The student acquires new vocabulary in each content area and uses it correctly.

National / Local Standards / Industry / ISTE:

UNDERSTANDINGS & GOALS

Enduring Understandings:

- Students will be able to discuss the importance of problem solving and logical thinking in Computer Science.

Essential Questions:

- What is an algorithm?
- How is problem solving used in computing?
- How can algorithms be evaluated?
- How can algorithms be adjusted?
- What is a problem statement?
- How is a flowchart used to solve problems?
- What is psuedocode and how is it used to develop solutions?

Knowledge from this Unit:

- Students will be able to communicate the design of an algorithm in both verbal and written form.
- Students will demonstrate the ability to use object-oriented programming concepts to create programs.
- Students will use procedures to solve problems (including procedures with parameters).
- Students will use basic programming constructs, including conditional statements, iteration, and variables to solve problems.

Skills from this Unit: Performance.

- Students will debug and test programs.

ASSESSMENT(S)

Assessment Method Type: Select one or more of the following. Please consider the type(s) of differentiated instruction you will be using in the classroom.

- ____ Pre-test
- ____ Objective assessment - multiple-choice, true- false, etc.
  - __ Quizzes/Tests
  - __ Unit test
- ____ Group project
- x Individual project
- ____ Self-assessment - May include practice quizzes, games, simulations, checklists, etc.
  - __ Self-check rubrics
  - __ Self-check during writing/planning process
  - __ Journal reflections on concepts, personal experiences and impact on one's life
  - __ Reflect on evaluations of work from teachers, business partners, and competition judges
  - __ Academic prompts
  - __ Practice quizzes/tests
- x Subjective assessment/Informal observations
  - __ Essay tests
  - __ Observe students working with partners
  - __ Observe students role playing
- ____ Peer-assessment
  - __ Peer editing & commentary of products/projects/presentations using rubrics
  - __ Peer editing and/or critiquing
- x Dialogue and Discussion
  - __ Student/teacher conferences
  - __ Partner and small group discussions
  - __ Whole group discussions
  - __ Interaction with/feedback from community members/speakers and business partners
- ____ Constructed Responses
  - __ Chart good reading/writing/listening/speaking habits
  - __ Application of skills to real-life situations/scenarios
- ____ Post-test

Assessment(s) Title:

Scratch Project

Assessment(s) Description/Directions:

Use the directions and rubric below for the assessment.

Students will have the opportunity to practice presentation and public speaking skills by presenting their project.

Students will also have the opportunity to communicate interpersonally while discussing algorithms.

Attachments for Assessment(s):

Scratch website (free download) http://scratch.mit.edu/download

Scratch is an easy to use graphical tool that can be used to teach programming concepts to students. Scratch is free and can be downloaded at http://scratch.mit.edu/download. Once you have downloaded and extracted the

.jar file you are ready to begin. No installation is needed as students will just click on the Scratch icon. By using the materials provided you will cover the programming standard in Computing in the Modern World. You will likely find that students will quickly understand how to use Scratch.

LEARNING EXPERIENCES

Instructional planning: Include lessons, activities and other learning experiences in this section with a brief description of the activities to ensure student acquisition of the knowledge and skills addressed in the standards. Complete the sequence of instruction for each lesson/task in the unit.

Sequence of Instruction

Lesson 1

1. **Identify the Standards. Standards should be posted in the classroom for each lesson.**
2. **Review Essential Questions.**
    - What is an algorithm?
    - How is problem solving used in computing?
    - How can algorithms be evaluated?
    - How can algorithms be adjusted?
    - What is a problem statement?
    - How is a flowchart used to solve problems?
    - What is psuedocode and how is it used to develop solutions?
3. **Identify and review the unit vocabulary.**

Sequence of Instruction:

- **Day 1:** Introduce Scratch to students, allow them to follow along with the steps in the file titled *Getting Started with Scratch.* If they finish early encourage them to take a look at the files included with the Scratch downloaded. There are many short clips and movies that are included.
- **Day 2:** Students work through six of the Scratch cards. Students should follow the steps on the Scratch card and save their individual files when finished with each.
- **Day 3:** Students work through the remaining five Scratch cards.
- **Day 4:** Review the Programming Concepts in Scratch document with students (sequencing, conditionals, iteration, variables, etc). After reviewing each topic students will create a new example demonstrating that concept.

- **Day 5 - ?:** Students will create the Scratch Project and present to class. Use your judgment on how long to spend on the project. Teachers may want to have students sketch their idea on paper before beginning.

- Technology is used in this unit on a daily basis. Students will use the Internet to research and explore related websites. Programming tools and software will be used to develop programs. Students will work with audio and video files to possibly embed into their programs.

Attachments for Learning Experiences:

Notes & Reflections:

- **This is a shortened version of the Alice unit and uses a similar program called Scratch that is simple and easy to learn. Please review the Teacher Guide Below.**
- **In the Scratch folder you will find a number of files. First, take a look at the *Teacher File* folder. If you have never used Scratch before please read the three PDF files in the folder.**
- **Next, review the student files. A suggestion timeline for working with Scratch is below. You can either print the files or simply let students use them as an e-book.**
- **This unit can be modified in any way. Prior to beginning the unit, it is imperative that students are given an extensive introduction to algorithm development and problem solving. It will also be helpful for students to be placed into small "support" groups when using the programming tools. These groups will be an excellent first line of defense for students when they have questions.**

CULMINATING PERFORMANCE TASK (Optional)

Culminating Unit Performance Task Title:

This is a final project where students will create a program, movie, or interactive game in Scratch.

Culminating Unit Performance Task Description/Directions/Differentiated Instruction:

Using what you have learned about Scratch, please create a Scratch program with the following elements. The topic is up to you and it could be a game or story/movie. Be creative and have fun. If you are having difficulty thinking of a project visit http://scratch.mit.edu for ideas.

Elements to include:

- 3 Sprites
- A conditional statement (if statement)
- Iteration (a loop)
- A sound
- A sensing statement
- A key press of mouse interaction
- A button

Rubric for Performance Task:

Task	Does not meet expectations	Needs Improvement	Perfect!	Student Score	Teacher Score
Sprites	No sprites are included(0)	Less than three sprites are included (1)	Three sprites are included (3)		
Event is included (key press, mouse event)	No events are included(0)	Event is attempted but is not functional (5)	A working event is included (10)		
Iteration (Loop)	No loop is used(0)	Loop is used but not functional(5)	Loop is used and is functional(10)		
Conditional (If statement)	No If statement is used (0)	If statement is used but not functional (5)	If statement is used and is functional (10)		
Sensing Statement	No sensing statement (0)	Sensing statement is included but is not functional(5)	Sensing statement is included and functional (5)		
Sound Imported	No sound is used (0)		Sound is used (2)		
Button	No button is included (0)	Button is included but doesn't work properly (5)	Button is included and works properly (10)		
			Total Points (out of 50)		

Title of the Student Work Sample

Embedded files in Scratch or visit http://scratch.mit.edu/ for examples.

Student Work Sample Description

A number of embedded files are in Scratch and the website has a gallery of student projects. Share these with students to generate ideas for their final project.

Attachments for Culminating Performance Task:

UNIT RESOURCES

Web Resources:

Attachment(s):

Materials & Equipment:

Computer

Scratch

Microsoft Word

LCD projector

Paper

What 21st Century Technology was used in this unit:

X	Slide Show Software		Graphing Software	X	Audio File(s)
	Interactive Whiteboard		Calculator		Graphic Organizer
	Student Response System	X	Desktop Publishing	X	Image File(s)
	Web Design Software		Blog		Video
X	Animation Software		Wiki		Electronic Game or Puzzle Maker
	Email	x	Website		