Summit Public Schools

Summit, New Jersey
Grade Level: 7 / Content Area: Robotics & Coding Cycle / 6 Weeks

Overview:

Robotics & Coding 7 is a cycle course that is transitioning to meet the needs of an ever changing technological world. It will serve as a bridge between the basics of Python presented in 6th grade and more advanced Python on the computers along with robotics in 7th grade. The students will be mixing text-based coding with robotics by using Python to program drones. The hands-on activities or challenges will vary in level of complexity depending on the previous experiences of the students. During the cycle, the students will use various data and design applications to enhance the learning and connect it to real life experiences.

Unit 1: Basic Python Programming
The unit will review with the students the concept of coding with text instead of visuals or blocks. The students will be learning the fundamentals of the specific programming language called Python. This is so the student will be able to use the basics in order to write simple code or programs using proper syntax.

Unit 2: Exploration with Drones
The unit will not only introduce the students to drones and their real world applications but also reinforce the Python programming skills that they have been learning. They will have the opportunity to apply these skills to robotics to see their programming in action in a 3D space rather than on a computer screen. Additionally, the students will be introduced to information about the commercial use of drones and make decisions about how they have impacted society as a whole. Students will also have the opportunity to make decisions about the usefulness of drones in a variety of scenarios through a series of challenges.

Unit 1: Basic Python Programming

Big Ideas: *Course Objectives/Content Statement(s)*

- To reinforce and advance the students' skills and knowledge of concepts in the Python Programming language.
- Computer Science is a field of study that heavily impacts our everyday life and the world simultaneously.

Essential Questions *What provocative questions will foster inquiry, understanding, and transfer of learning?*	**Enduring Understandings** *What will students understand about the big ideas?*
• What is Computational Thinking? • How do you program in Python? • What skills, concepts, and special functions are required to know in order to perform basic Python programming? • What is a string in Python and how can it be modified to desired outputs? • How can user input be incorporated into a program to generate various outputs? • Which coding skills and concepts can be learned through hands-on coding projects and activities? • How can students use their own coding skills and mindset to learn how to troubleshoot or debug any programming issues that arise? • How can SEL become a part of the students' coding experiences? • How can student individuality and creativity be expressed through coding?	• Computational Thinking is thinking like a computer and using the four steps to break down a problem. (Decomposition, Pattern Recognition, Abstraction, and Algorithms.) • Identify and explain the basic components of a Python program (e.g., variables, data types, operators). • Write and debug simple Python programs using correct syntax and indentation. • Apply basic control structures (e.g., conditionals, for loops) to solve programming problems. • Demonstrate an understanding of algorithmic thinking and problem-solving strategies. • Learn how to read and interpret error messages and use them to solve issues in the programming. • Build sustained focus and perseverance in order to debug a program. • Learn how to read and interpret error messages and use them to solve issues in the programming. • Design code or programs that reflect student interests while applying fundamental concepts and skills.
Areas of Focus: Proficiencies **(New Jersey Student Learning Standards)**	**Lessons**

Students will: **NJ Standard 8.1 - Computer Science** 8.1.8.AP.2: Create clearly named variables that represent different data types and perform operations on their values. **NJ Standard 8.1 - Computer Science** 8.1.8.AP.3: Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals. **NJ Standard 8.1 - Computer Science** **Disciplinary Concept: Algorithms & Programming** 8.1.8.AP.4: Decompose problems and sub-problems into parts to facilitate the design, implementation, and review of programs. **NJ Standard 8.1 - Computer Science** **Disciplinary Concept: Algorithms & Programming** 8.1.8.AP.5: Create procedures with parameters to organize code and make it easier to reuse. **NJ Standard 8.1 - Computer Science** **Disciplinary Concept: Algorithms & Programming** 8.1.8.AP.7: Design programs, incorporating existing code, media, and libraries, and give attribution. **NJ Standard 8.1 - Computer Science** **Disciplinary Concept: Algorithms & Programming** 8.1.8.AP.8: Systematically test and refine programs using a range of test cases and users. **NJ Standard 8.1 - Computer Science** **Disciplinary Concept: Algorithms & Programming** 8.1.8.AP.9: Document programs in order to make them easier to follow, test, and debug. **ISTE Standard 1.1. Empowered Learner:** 1.1.c Students use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.	**Lesson 1:** ● What is Computational Thinking? Vocabulary & Activity ● Review the concept of an algorithm and discuss the importance of pseudocode. **Lesson 2:** ● Explain and express a rudimentary understanding of basic Python syntax through simple commands. (Review) ● Identify and explain what is Input and Output. **Lesson 3:** ● Apply conditionals to detect whether a condition is true or can only run in certain cases. ● Experiment with assignment operators, comparison operators, and logical operators using basic programming. **Lesson 4:** ● Discuss the concept of loops and various types. ● Compare/contrast loops in block coding vs. text-based coding. ● Identify the difference between "for" loops and "while" loops ● Write a basic program using loops to demonstrate how they work in text-based coding. **Lesson 5:** ● Discuss the use of variables and the different types of data that can be stored in them. ● Use proper syntax for strings and explore the concept of concatenation. ● Write basic code/programs that apply the different types of data that can be used in a variable and the different operations that can be applied to a variable.

1.1.d Students understand the fundamental concepts of technology operations, demonstrate the ability to choose, use and troubleshoot current technologies and are able to transfer their knowledge to explore emerging technologies. **ISTE Standard 1.2. Digital Citizen:** 1.2.b Students engage in positive, safe, legal and ethical behavior when using technology, including social interactions online or when using networked devices. **ISTE Standard 1.5. Computational Thinker:** 1.5.d Students understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions. **ISTE Standard 1.6. Creative Communicator:** 1.6.b Students create original works or responsibly repurpose or remix digital resources into new creations. 1.6.c Students communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations, models or simulations. **Career-Ready Practices** **CRP1**: Act as a responsible and contributing citizen and employee. **CRP2**: Apply appropriate academic and technical skills. **CRP3**: Attend to personal health and financial well-being. **CRP4**: Communicate clearly and effectively and with reason. **CRP5**: Consider the environmental, social and economic impacts of decisions. **CRP6**: Demonstrate creativity and innovation. **CRP7**: Employ valid and reliable research strategies. **CRP8**: Utilize critical thinking to make sense of problems and persevere in solving them. **CRP9**: Model integrity, ethical leadership and effective management. **CRP10**: Plan education and career paths aligned to personal goals. **CRP11**: Use technology to enhance productivity. **CRP12**: Work productively in teams while using cultural	

global competence.

Differentiation

Interdisciplinary Connections

- Science & Math - Coding falls under STEM. Students need to use concepts from Math and Science in order to code.

Technology Integration

- Python & IDE (PyCharm or equivalent)
- Desktop computers with internet connection

Media Literacy Integration

- Articles and Instructional videos

Global Perspectives

- Students will potentially see instructional videos that not only talk about how they can use coding but how others do as well. These videos will introduce the students to the perspective of various coders in different professional roles.

Supports for English Language Learners

Sensory Supports	Graphic Supports	Interactive Supports
Real-life objects	Charts	In pairs or partners
Manipulatives	Graphic Organizers	In triads or small groups
Pictures	Tables	In a whole group
Illustrations, diagrams & drawings	Graphs	Using cooperative group
Magazines & Newspapers	Timelines	Structures
Physical activities	Number lines	Internet / Software support
Videos & Film		In the home language

Assessments

Formative Assessments:

- Practice Exercise with basic syntax and commands
- Practice Exercise with variables
- Practice Exercise with loops

Summative Assessments, Projects, and Celebrations:

- Students created basic Python (text-based code) programs that demonstrate the required skills and concepts learned through the lessons.

Broadcasts		With mentors
Models & Figures		

Intervention Strategies		
Accommodations	**Interventions**	**Modifications**
Allow for verbal responses	Multi-sensory techniques	Modified tasks/expectations
Repeat/confirm directions	Increase task structure (e.g. directions, checks for understanding, feedback	Differentiated materials
Permit response provided via computer or electronic device	Increase opportunities to engage in active academic responding	Individualized assessment tools based on student need
Audio Books	Utilize pre-reading strategies and activities previews, anticipatory guides, and semantic mapping	Modified assessment grading

Recommended Texts to Support Unit:

Unit 2: Exploration with Drones

Big Ideas: *Course Objectives/Content Statement(s)*

- To introduce the students to Text-Based Coding through the use of drones or UAV.

Essential Questions *What provocative questions will foster inquiry, understanding, and transfer of learning?*	**Enduring Understandings** *What will students understand about the big ideas?*
• What is a drone or a UAV (Unmanned Aerial Vehicle)? How is a drone able to fly and stay in the air? • What are the main components of a drone or UAV? • Are there key principles, terms, and ideas that need to be understood in order to operate a drone? • How does one pair, fly, and program a drone to perform an autonomous task? • What safety protocols are there for operating a drone? • How are drones used in a real world setting? • How can students use their own coding skills and mindset to learn how to troubleshoot or debug any programming issues that arise?	• The basic concept of what a drone or an UAV is and be able to identify the main components. • Basic operation and procedures for flying safely. • Safe drone operation requires following pre-flight checks and adhering to responsible flying practices. • Drones must be paired in order to receive python commands. • Basic Python code can be used to send control signals and automate drone movements. • Flight terminology such as trim, pitch, roll, and yaw) • Understand how to use the takeoff(), land(), and hover(int) functions to navigate with your drone. • Drones have the potential for various applications, especially in data collection, delivery service and search and rescue operations. • Learn how to read and interpret error messages and use them to solve issues in the programming.
Areas of Focus: Proficiencies **(New Jersey Student Learning Standards)**	**Lessons**
Students will: **NJ Standard 8.1 - Computer Science** **Disciplinary Concept: Computing Systems** 8.1.8.CS.4: Systematically apply troubleshooting strategies to identify and resolve hardware and software problems in	**Lesson 1:** • General introduction to drones & discussion on how a drone is able to fly by using the right commands in their programming.

computing systems.

NJ Standard 8.1 - Computer Science
Disciplinary Concept: Algorithms & Programming
8.1.8.AP.1: Design and illustrate algorithms that solve complex problems using flowcharts and/or pseudocode.

NJ Standard 8.1 - Computer Science
Disciplinary Concept: Algorithms & Programming
8.1.8.AP.2: Create clearly named variables that represent different data types and perform operations on their values.

NJ Standard 8.1 - Computer Science
Disciplinary Concept: Algorithms & Programming
8.1.8.AP.4: Decompose problems and sub-problems into parts to facilitate the design, implementation, and review of programs.

NJ Standard 8.1 - Computer Science
Disciplinary Concept: Algorithms & Programming
8.1.8.AP.5: Create procedures with parameters to organize code and make it easier to reuse.

NJ Standard 8.1 - Computer Science
Disciplinary Concept: Algorithms & Programming
8.1.8.AP.7: Design programs, incorporating existing code, media, and libraries, and give attribution.

NJ Standard 8.1 - Computer Science
Disciplinary Concept: Algorithms & Programming
8.1.8.AP.8: Systematically test and refine programs using a range of test cases and users.

NJ Standard 8.1 - Computer Science
Disciplinary Concept: Algorithms & Programming
8.1.8.AP.9: Document programs in order to make them easier to follow, test, and debug.

ISTE Standard 1.1. Empowered Learner:
1.1.c Students use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways.

- Drones in the World - Webquest
- Discussion that focuses on how drones can be used to address real-world challenges and improve our quality of life.

Lesson 2:
- Getting to know your drone & basic drone terminology
- Drone Safety

Lesson 3:
- Pairing of the drone in preparation for programming.
- Apply the coding commands in order to takeoff, hover, and land.
- Complete a challenge where the students apply their knowledge of these concepts using the drone.

Lesson 4:
- Learn what the flight directions (throttle, pitch, yaw, and roll) are and how to program them in Python.
- The students will then apply what they have learned to have their drone fly through an obstacle course to represent a flight pattern.

Lesson 5:
- Apply their knowledge of "for" loops, "while" loops, and conditionals to write and rewrite code more efficiently when flying a drone.
- Complete a challenge where the students will apply loops to their programming in order to make the drone fly in a repetitive motion.

Lesson 6:
- Learn how to select and program the LEDs to light up.
- Program the drone to display the LED lights in a student designed sequence.

1.1.d Students understand the fundamental concepts of technology operations, demonstrate the ability to choose, use and troubleshoot current technologies and are able to transfer their knowledge to explore emerging technologies. **ISTE Standard 1.2. Digital Citizen:** 1.2.b Students engage in positive, safe, legal and ethical behavior when using technology, including social interactions online or when using networked devices. **ISTE Standard 1.3. Knowledge Constructor:** 1.3.d Students build knowledge by actively exploring real-world issues and problems, developing ideas and theories and pursuing answers and solutions. **ISTE Standard 1.4. Innovative Designer:** 1.4.a Students know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts or solving authentic problems. 1.4.c. Students develop, test and refine prototypes as part of a cyclical design process. 1.4.d Students exhibit a tolerance for ambiguity, perseverance and the capacity to work with open-ended problems. **ISTE Standard 1.5. Computational Thinker:** 1.5.d Students understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions. **ISTE Standard 1.6. Creative Communicator:** 1.6.b Students create original works or responsibly repurpose or remix digital resources into new creations. 1.6.c Students communicate complex ideas clearly and effectively by creating or using a variety of digital objects such as visualizations, models or simulations. **Career-Ready Practices** **CRP1**: Act as a responsible and contributing citizen and employee. **CRP2**: Apply appropriate academic and technical skills.	

CRP3: Attend to personal health and financial well-being. **CRP4**: Communicate clearly and effectively and with reason. **CRP5**: Consider the environmental, social and economic impacts of decisions. **CRP6**: Demonstrate creativity and innovation. **CRP7**: Employ valid and reliable research strategies. **CRP8**: Utilize critical thinking to make sense of problems and persevere in solving them. **CRP9**: Model integrity, ethical leadership and effective management. **CRP10**: Plan education and career paths aligned to personal goals. **CRP11**: Use technology to enhance productivity. **CRP12**: Work productively in teams while using cultural global competence.	
Differentiation	**Assessments**
Interdisciplinary Connections **Math:** ● Understanding (x,y,z) coordinates is crucial for programming drone movements. ● Visualizing drone positions and translating them into code requires spatial reasoning. **Science:** ● Understanding forces of flight such as lift, thrust, drag and gravity are essential for controlling drone flight. **Technology Integration** ● Desktop Computers with Internet Connection ● Python (Program/Application); if applicable ● PyCharm (Code Editor); if applicable ● Instructional videos - Youtube, etc **Media Literacy Integration** ● Newsela **Global Perspectives** ● The students will explore how drones are used in a variety of ways that benefit human society. Topics	**Formative Assessments:** ● Teacher Observations ● Practice Exercises with each of the concepts presented in the individual lessons ● Student Reflections on tasks and challenges **Summative Assessments, Projects, and Celebrations:** ● Drones in the World Webquest ● Current Events Articles & Assignments ● Drone Challenges & Flight exercises

will range from everyday assistance to search and rescue efforts.

Supports for English Language Learners		
Sensory Supports	**Graphic Supports**	**Interactive Supports**
Real-life objects	Charts	In pairs or partners
Manipulatives	Graphic Organizers	In triads or small groups
Pictures	Tables	In a whole group
Illustrations, diagrams & drawings	Graphs	Using cooperative group
Magazines & Newspapers	Timelines	Structures
Physical activities	Number lines	Internet / Software support
Videos & Film		In the home language
Broadcasts		With mentors
Models & Figures		

Intervention Strategies		
Accommodations	**Interventions**	**Modifications**
Allow for verbal responses	Multi-sensory techniques	Modified tasks/expectations
Repeat/confirm directions	Increase task structure (e.g. directions, checks for understanding, feedback	Differentiated materials
Permit response provided via computer or electronic device	Increase opportunities to engage in active academic responding	Individualized assessment tools based on student need

Audio Books	Utilize pre-reading strategies and activities previews, anticipatory guides, and semantic mapping	Modified assessment grading	

Recommended Texts to Support Unit: