SCHOOL of EDUCATION

INSTRUCTIONAL PLAN

Name: Gary Malkasian	**Grade: 9-12**
Date: 2/4/25	**Unit/Subject: Computer Science: Programming in C#**
Lesson: Loops and Counters	**Number if Part of a Series Lessons: _ of _**

Learning Target(s)

Long-term Learning Goal(s) (i.e., teacher goal for unit or learning segment): Students will learn computer science loop theory with both basic and advanced techniques, and when it is best to use each type. Students will translate a spoken algorithm using loops into code.

This lesson:
Create programs that use for-loops.
Create programs that use while-loops.
Create programs that use do-while-loops.
Design programs that use nested loops.
Determine when it is best to use each type of loop.
Use a string as a character array
Create a computer program to solve a problem

Common Core: Reading Standards for Literacy in Science and Technical Subjects 6–12
Key Ideas & Details
Follow precisely a complex multistep procedure when carrying out experiments, taking measurements, or performing technical tasks, attending to special cases or exceptions defined in the text.

Craft & Structure
Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to grades 9–12 texts and topics.

OPSI Computer Science K-12 Standards
1B-AP-10 Create programs that include sequences, events, loops, and conditionals.
2-AP-11 Create clearly named variables that represent different data types and perform operations on their values.
2-AP-12 Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.

Learning Target(s) for this Lesson: (in kid-friendly language): SW learn the different types of loops and the best time to use each type. SW use loops to solve a programming problem.

Academic Language explicitly taught in this lesson – vocab, language function (processes such as explain, describe, analyze), syntax, discourse

Academic Language taught: increment, decrement, prefix, postfix, pretest, posttest, cycling, declaration

Academic Language Supports (charts, handouts, videos etc.): PowerPoint, sample code

Personal, Cultural or Community Asset Connections

Plan (what will I do to connect the target(s) to students and how?):
The lesson will include an assignment to create a program that can detect a palindrome. Palindromes are fun and interesting. There are many well-known and humorous ones.

Background Knowledge

How will I activate background knowledge? The students have already created several programs with loops so they are familiar with the basic concept.

How will I fill prior knowledge gaps? Students have worked with loops but not all types of loops, how they work, and when to use each type. They may not know more advanced techniques. And some have blindly used them following instructions without really understanding how they are structured and why. The PowerPoint contains graphical analogies to explain looping behavior. Much of the lecture will be review, but with enough advanced material to interest students who are not struggling. During the lab portion, I will go around the room giving feedback, hints, and corrections to their code as needed.

Addressing "the hard part"

What will be the "hard part" for kids about this learning target(s)? Possible misconceptions?
This is new syntax that simply has to be learned. The concepts of "for loop", "while loop", and "do while loop" may seem foreign. The student has to practice to become "fluent" with these expressions over time. Students will have time to create simple loops and experiment with how they behave.

How will I address that "hard part" (what will I do and how?) PowerPoint, instruction, examples, individual attention, coaching

Instructional Materials, Resources, and Technology Needed to Present this lesson

Large Screen for PowerPoint presentations and videos
Classroom computers

Internet access
Schoology

Instructional Plan to Deepen and Extent Student Understanding:

Teacher Tasks (steps of the lesson) and Student Learning Behavior (what you'll see)-Teacher Will or Student Will (TW or SW)- Start with an Action Word

TW greet students as they enter
After the bell, TW take attendance and introduce guest
TW introduce topic and state the learning objectives

TW give PowerPoint presentation, explaining how loops work within computer programs. At key points during presentation, TW conduct formative assessments, asking students to 1) give a fist-of-five, 2) state which concept is the muddies, or 3) turn-and-talk and then report their understanding of the material.

SW will open the Integrated Development Environment (IDE) application on their computer to create a simple program. Using what they've learned about loops, the students will write a program that will:

a) Ask the user to input a string from the keyboard
b) Assign the string to a variable
c) The program will test the string to determine if it is a palindrome
d) The program will print the results to the screen
e) The program will ask the user if they want to enter another string
f) If the user enters yes, program loops back to step a). If the user enters no, the program terminates.
g) Students are encouraged to customize their program with new questions and messages to the user. This is optional.

TW circulate around the room to help students
TW will give extra help to students that are struggling.
SW turn the completed project in through Schoology (Learning Management System).

The simplest implementation to code is to reverse the string with a decrementing for-loop, filtering out spaces and punctuation, and doing a case-insensitive comparison to the original string.

Challenge: A more efficient algorithm is to create a for-loop with two counters that work from both the start and end of the string and stop when they meet. The for-loop could have nested while-loops to filter out spaces and punctuation. It's okay if students don't complete the challenge assessment because we will have loops in nearly all assignments. For loops and nested loops will be revisited many times.

Accommodations/Modifications Needed for Individual or Diverse Learners

- Students will have a choice of methods to solve the coding problem. The easier method uses techniques the students have used the past. The challenge method includes new techniques and results in more efficient code.
- Priority seating for those requesting it.
- PDF of PowerPoint will be available for the students.
- Tutoring support for those who require it
- Will have ready skeleton program that students can modify, for those struggling to create it

Supports for specific special needs students for this specific target(s):

Students with SPED:
The c*hallenge* techniques can be omitted
TW circulate, giving them priority for help
TW assist creating a bare bones program
TW show student how to step through the code line-by-line and see it executing
Additional tutoring can be available during office hours/choice time or after school

Assessment Strategies: Assessing the Impact of the Instruction on Student Learning

Proposed evidence (Student work, skills, performance):	
Performance Assessment: students will create a program and submit that program to Schoology for credit. **1B-AP-10** Create programs that include sequences, events, loops, and conditionals. Program will contain at least one loop, a sequence of commands, logical flow control by conditional expressions, and handle keyboard events. **2-AP-11** Create clearly named variables that represent different data types and perform operations on their values. SW use Boolean logic in flow control and advanced string manipulation to decide if a given string is a palindrome.	Evaluation Criteria (What will tell me students have "met standard, etc.")? **Performance Assessment**: The evaluation criteria is a working program. A working program demonstrates: Student is able to successfully create loops in a program, including nested for- and while-loops Student is able to use logical controls within a program Student is able to manipulate strings

Challenge: 2-AP-12 Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals. SW use nested loops and compound conditionals in this program. Having an optional challenge exercise gives the opportunity to work iteratively.	
Plan to give students feedback: Formative Assessment: TW ask students to fist-of-five, muddiest concept, and turn-and-talk at key points during the PowerPoint presentation to verify understanding, and give clarifications, as needed. For Performance Assessment: TW circulate around the room and work with students, giving praise for well written algorithms, pointing out syntactic and logical errors, and helping the student fix the errors, as needed. If at the end of the class period a large percentage of the class has not yet successfully created a working program, we will dedicate more class time to give more instruction, clarification, and tutoring and to finish the project in the next class.	How student will use your feedback: SW will get clarifications during PPT presentation based on results from formative assessments specified above. SW revise work until they have a working program. Student will build knowledge and confidence in working with loops. For most students, it is exhilarating to see a program they wrote executing and doing what they wanted it to do.