Getting started with Micro:bit

Directions: Use this planning worksheet to follow the three steps of the backwards design process in order to plan an effective lesson.

Subject Computer Science	**Lesson Date:** 12-4-21
Content Standard Math: Problem solving Computer Science: 5.CS.D.01 Computing Systems: Devices 5.CS.HS.01 Computing Systems: hardware and Software 5.AP.A.01 Algorithms and Programing: Algorithms 5.AP.V.01 Algorithms and Programing: Variables 5.AP.C.01 Algorithms and Programing: Control 5.AP.PD.01 Algorithms and Programing: Program Development 5.IC.SI.0 Impacts of Computing: Social Interactions Technology ISTE: 3a, 3c, 5b, 7a	

Step One: Write a Student-Centered Learning Objective – Must be specific, measurable, and clearly stated.

Behavior – WHAT the learner will be able to do. Includes a verb!	I can complete a specific programming task on a Micro:bit by observing tutorials and working collaboratively with my partner.
Condition – HOW the learner will perform the behavior. Refers to a tool, reference, aid, or context they will or will not be able to use.	Students will learn introductory programming codes for Micro:bits using tutorial videos found on makecode.mirobit.org, and direct instruction from the teacher.
Criterion – How WELL the learner must perform to demonstrate content mastery. Refers to a degree of accuracy, number of correct responses, or time imit.	Below Basic- 0-1 One or two codes imputed, but the task (eg. making dice/name tag) is less than 50% complete Basic- 2 Three to Four codes imputed, but the task (eg. making dice/name tag) is less than 75% complete Proficient- Five or more codes imputed, and the task (eg. making dice/name tag) is at least 90% complete Advanced- 4 Five or more codes imputed, and the task (eg. making dice/name tag) is 100% complete, plus additional steps/codes added

Learning Objective – Put all three parts together.	I will learn about programming a Micro:bit through exploration of tutorial videos. I will program the Mocro:bit to perform as dice and/or name tags

Step Two: Create a Plan for Assessment – Used to gather information about a student's progress towards mastery of the learning objective, help the teacher identify what instruction is working well and what needs refinement, and informs the students about their learning.

Type of Assessment	Options to Consider	Specific Plan
Diagnostic / Pre-Assessment – Used to check prior knowledge before a lesson.	☐ Self-Assessment ☐ Writing Prompts ☐ Running Records ☐ Performance Task ☒ Other	Micro:bit observation and group discussion.
Formative – Used during a lesson to check progress, identify any misconceptions, and give feedback to students.	☐ Learning / Response Log ☐ Admit / Exit Ticket ☐ Think / Pair / Share ☐ One Minute Paper ☒ Other	Work as a pair/team to complete the task of programming a Micro:bit to perform as dice and/or name tags.
Summative – Used at the end of a lesson to check student mastery of the objective.	☐ End of Unit Tests ☒ Final Exams or Mid-Term Exams ☐ State Tests ☐ Portfolio ☐ Project	Give a written response of how the programming took place, including as many details of the codes as possible, and a reflection of how well (+/-) the team worked to complete the task/s.

Step Three: Choose Learning Strategies and Activities – How you present new content to your students, and how your students will actually interact with the content. Add additional rows as needed.

Strategy 1: Anticipatory Set ☐ Direct Teach ☐ Demonstration ☐ Cooperative Learning ☒ Discover /Inquiry-Based Learning ☐ Project-Based Learning ☐ Other: _____	Activities Planned: ☒ Active ☐ Passive Teams will make observations of the Micro:bit and predict what it could be used for.
Strategy 2: ☒ Direct Teach ☒ Demonstration ☐ Cooperative Learning ☐ Discover /Inquiry-Based Learning ☒ Project-Based Learning ☐ Other: _____	Activities Planned: ☒ Active ☒ Passive Teams will learn introductory programming codes for Micro:bits using tutorial videos found on makecode.mirobit.org, and direct instruction from the teacher, while completing tasks.