Secondary / ESL Lesson Plan Template -- 2021-2022 Academic Year

Name: Nora Evans	Lesson Topic: Recursion
Content Area: Computer Science/Computer Science 1	Grade Level(s): 9-12

Lesson Content

Background Information/ Relevance/ Context/ Rationale (Purpose) *– Please be clear about how this particular lesson is situated within the current instructional sequence (i.e., unit), why this content is important for students to learn, and how you will convey the relevance and significance of this lesson to students.*

Computer Science 1 is a self-paced class that follows the CS Academy CS1 curriculum. The students must complete the first four units in the first semester. In unit 1, students learned to create drawings, in units 2 and 3 learned event driven programming concepts. Most of the students are learning section 4, conditional statements. In addition to the online course, students must complete in-class activities that teach computational and algorithmic thinking.

In this class, the students continue to work on the online programming course. In addition, the students will understand the meaning of recursion through problem solving. The students identify smaller instances of the problem in a solution. Recursion is a powerful problem-solving technique that can be easily implemented in computer code.

VSOLs/CCSSs Addressed:

CSF.12 The student will develop a program working individually and in teams using a text-based language.
CSF.13 The student will identify the expected output of a program given a problem and some input.
~~CSF.14 The student will design and iteratively develop programs for practical intent or personal expression, incorporating feedback from users.~~
CSF.15 The student will design and implement algorithms using
a. sequencing of instructions;
b. conditional execution; and
c. iteration.
~~CSF.16 The student will implement a program that accepts input values, stores them in appropriately named variables, and produces output.~~
~~CSF.17 The student will trace the execution of an algorithm, illustrating output and changes in values of named variables.~~
~~CSF.18 The student will apply the basic operations used with numeric and non-numeric data types in developing programs.~~
CSF.19 The student will use predefined functions to simplify the solution of a complex problem.
CSF.20 The student will apply simple algorithms to a collection of data.
~~CSF.21 The student will create programs~~
~~a. demonstrating an understanding that program development is an ongoing process that requires adjusting and debugging along the way; and~~
~~b. using version control to create and refine programs.~~

Learning Targets

As a result of this lesson, students will...

Explore the following ESSENTIAL QUESTIONS…	**UNDERSTAND THAT…**
(EQ1) How do we solve complex problems?	(U1) The students will understand that recursion is a technique that uses solutions of smaller problems to solve the bigger problem.
KNOW…	**BE ABLE TO (DO)...**
(K1) The students will know that recursion is a widely used problem solving technique.	(D1) Solve the Towers of Hanoi problem for 3, and 4 disks using a wooden puzzle. (D2) Calculate the smallest number of moves needed to solve the puzzle.

Plan for Assessment

As you plan your lesson, think about what evidence you will gather to determine if students have mastered the lesson's learning targets. Develop a plan for assessing the degree to which your students have mastered the learning targets in this lesson. ***Your plan should include at least one formative assessment at a minimum****, and may also include diagnostic/pre-assessment or summative assessments depending on the nature of the learning targets and the placement of the lesson within the context of the unit.* ***Please add/remove tables as needed.***

Assessment #1

Learning target(s) for which you will collect data (write out):

(U1) The students will understand that recursion is a technique that uses solutions of smaller problems to solve the bigger problem.

Type of Assessment (choose and highlight one):	**Possible Options:**	
• Diagnostic Assessment or Pre-Assessment (before the lesson) • Formative Assessment (during the lesson) • Summative Assessment (after the lesson)	• Previous class work • Graphic organizer • Journal entry • Think/Pair/Share • 1-minute essay	• Problem set • Strategic questioning • Learning/response log • Exit ticket • 3-2-1 countdown • Other (DO NOW)

Specific Plan *(What does the assessment entail, and what evidence will help you determine students' progress towards the learning target(s)?)*:

The assessment is used as a DO NOW activity to connect the previous lesson on binary search to recursion.
The students record the answers on a worksheet provided by the teacher. The teacher collects the papers, analyses the results and hands back the worksheet in the next class.

The teacher collects evidence of the students' understanding of the binary search strategy and their ability to apply it in a novel situation.
If a student does not split the list consistently into two sub-lists of equal length or length different by 1, then it is likely that the student does not understand the binary search strategy or is incapable of determining the middle of an interval.
More specifically, if a student can not complete the first two steps, it is likely the student does not know how to use the binary search strategy. If a student is able to complete the first 2-3 steps, then the student is likely able to apply the binary search strategy to the problem studied in the previous class, but is unable to apply it to a new problem.

Steps in the Lesson (including Modalities & Resources)	**Planning for Learner Diversity / Instructional Scaffolds** ● You should identify specific ways you plan to ensure equity and inclusion by building on learners' diversity. ● Guiding questions: ○ *How can you leverage students' prior experiences and assets in this step of the lesson?* ○ *How does this step address and build on learner diversity to ensure equity?* ○ *What challenges/ opportunities/barriers might exist for individuals or groups of learners in accessing content and learning activities in this step of the lesson?* ○ *How will you utilize Universal Design for Learning (UDL) and/or differentiation so that all students can access the content and learning activities?* ○ *What feedback can you provide to acknowledge and value each student's contributions?*

Steps in the Lesson (including Timing and Modalities & Resources Utilized)	*Planning for Student Diversity / Instructional Scaffolds*
STEP 1 The teacher greets the students at the door and hands-out the worksheets. The lesson schedule is projected on the whiteboard: 1. Entry ticket and self-paced work (30 minutes) 2. Lesson: Towers of Hanoi (30 min) 3. Self-paced work (20 minutes)	
STEP 2: Entry ticket and self-paced work **The teacher says:** *Please complete the entry ticket at your leisure within the next 30 minutes. The entry ticket is independent work, but you may ask clarifying question from the teacher or classmates.* *In the rest of the time, please keep working on the CS Academy exercises.* *The lesson notes from the last activity are available in Schoology.* *To make it easier for you to complete the optional coding activities, I created Jupyter notebooks for the in-class activities.* The teacher shows the students how to access the Jupyter notebook.	For the students that have difficulties with the concept or that are interested in an alternate presentation of the binary search, there is an optional activity on brilliant.org For the students that are interested in programming the binary search, the Jupyter notebook allows them to write and check their code.
STEP 3: Lesson The teacher fills in the correct answers in the Entry Ticket with the students' help. **The teacher says:** *Today, we explore the concept of recursion using a mathematical puzzle, called the Towers of*	If a group falls too far behind, they may skip the 4-disks problems. If a group finishes quickly and is interested in doing the optional activities, they may start before the rest of the class

Hanoi. You will work in eight groups of 3-4 students to solve the puzzle I will hand you in a minute. Please form groups and elect a leader responsible with collecting the puzzle and returning it at the end of the activity. (3 minutes) *I explain the rules of the puzzle. You may follow along in the Jupyter notebook if you wish. The game starts with a set of disks stacked in decreasing order of size with the largest on the bottom. The goal is to move the entire stack from the rod labeled A to the one labeled B. You may only move one disk at a time, and you may placed a disk only on top of a larger disk or empty rod.* The teacher demonstrates the moves for 2 disks and explains the table that records the moves. *Now, it is your turn to solve the puzzle for three disks. Please record the moves, either in the Jupyter notebook or a piece of paper. (4 minutes)* *Once you finish recording the moves, please complete the reflection.* The teacher demonstrates the correct solution and identifies the 2 size 2 problems. *Please solve the puzzle for 4 disks, respond to the reflection questions.* The teacher monitors the progress of the students. If several groups finish quickly and other groups struggle, then the teacher announces that the groups that completed the activity may move on to the next section. If no groups finish quickly, the teacher solves the puzzle with the class and guides the class through the next section. *The problem-solving technique you used in this puzzle is called recursion. To solve the 3-disk problem, we solved the 2-disk problem, made a move, and then solved the 2-disk problem again. The same was for 3 and 4 disks: solve the problem for one less disk, make a move, solve the problem for one less disk.* *When do we stop? We stop when the problem is so simple, we solve it in one step.* *Here is the Python code. Would you like to discuss it as a class, or do you prefer to inspect it in your own time?* The teacher follows up accordingly. *AT the end of the lesson, you can find optional activities to do is*	finished the 4-disks problem.
STEP 4: Self-paced work The teacher moves around the classroom and answers the students' questions.	For the students that need extra help understanding the programming exercises, the teacher uses a portable whiteboard to draw diagrams of the programming challenges. If a student struggles to understand Python error messages, the teacher helps the student decode them and guides the student to fixing the error. Since this is a self-paced course, students that find it easy move to next units.

STEP 5: End of class	

Materials/ Supplies/ Sources/ Digital and Interactive Instructional Technology (if appropriate): – Please list all necessary instructional supplies, materials, and sources. Make sure that these are clearly labeled and referenced throughout the lesson plan to enhance clarity.

Entry Ticket (see attached EntryTicket.docx file)
Towers of Hanoi Wood Puzzle (https://www.amazon.com/gp/product/B095STN3J6/ref=ppx_yo_dt_b_search_asin_title?ie=UTF8&psc=1)
Chromebooks (school-provided computers)
Brilliant.org subscription (free for teachers)
Instructions for using Jupyter notebooks https://github.com/ananevans/MonticelloHS-CS1
Towers of Hanoi notebook https://github.com/ananevans/MonticelloHS-CS1/blob/main/02_TowersOfHanoi.ipynb