**AP CSP CodeBot**

|  |  |  |
| --- | --- | --- |
| **LESSON: Traversing a List** | | **Time: 45 minutes** |
| **Project Goal:** Students will traverse a single list and multiple lists in a program.  **Learning Targets**   * I can define a list of different data types. * I can traverse a list using a specialized for loop. * I can traverse multiple lists using a for loop. * I can traverse a list multiple times without getting an index out of range error. | **Key Concepts**   * Traversing a list means accessing each element in a list, in order. * The easiest way to traverse a list is to use a for loop. * To traverse a single list one time, use a specialized for loop where the counter variable holds each value of the list. * You can traverse multiple lists at the same time, and traverse a list multiple times using a for loop. | |
| **Assessment Opportunities**   * Traversing a List CodeBot Assignment * PythonLists2 program * LineSense\_lists program | **Success Criteria**   * Define a list with a data type other than integers * Traverse a list using a specialized for loop * Traverse two lists using a for loop * Traverse multiple lists multiple times using a for loop and the modulo operator | |
| **AP CSP Framework**  **AAP-1.C** Represent a list or string using a variable.  **AAP-2.N** Write expressions that use list indexing and list procedures.  **AAP-2.O** Write iteration statements to traverse a list.  **Computational Thinking Practice 3.A** Generalize data sources through variables.  **Computational Thinking Practice 4.C** Identify and correct errors in algorithms and programs, including error discovery through testing. | **Materials**   * Traversing a List slides * Traversing a List CodeBot Assignment / Answers * Unit 4 Review and Test Questions * Code solution for PythonLists2 * Code solution for LineSense\_lists | |
| **Teacher Notes**   * This lesson will be completed on the computer, using CodeSpace for programming. The CodeBot will still need to be connected in order to run code. * Use the Sandbox in CodeSpace for programming. This lesson is not part of a mission. * The assignment can be distributed digitally. Space is provided for students to take notes during the programming. * Students will add to the PythonLists1 program, and also modify the LineSense program. * The best experience will come from them modifying their own code. However, we want all students to be engaged, so you can give them the original code to modify if needed. * The most recent version of each program can be found in the earlier assignments. If you are giving code to students, use the solution code found here:   + PythonLists1: Lists with CodeBot Lesson   + LineSense\_final: Mission 5 Obj 6-9 * Follow the slides for instructions and guidance. * Solution code for PythonLists2 and LineSense\_lists are provided. | | |