**AP CSP CodeBot**

|  |  |  |
| --- | --- | --- |
| **MISSION 6 Obj 7-8 Line Follower** | | **Time: 45 minutes** |
| **Project Goal:** Students will use line sensor input to program the ‘bot to follow a line.  **Learning Targets**   * I can use tuples to create proportional speed control. * I can use built-in math functions to calibrate the CodeBot. | **Key Concepts**   * Auto-calibration algorithms can make your ‘bot adaptable to any environment. * Global variables exist outside a function, while local variables are created inside a function. You can declare a variable global in a function. * Python has many built-in functions formath. | |
| **Assessment Opportunities**   * Mission 6 Obj 7-8 Assignment * Submit the “LineFollower” program (final) * [Mission 6 Kahoot Review](https://create.kahoot.it/share/firia-labs-codebot-mission-5/20d9499d-fe50-45a7-9f2c-623975832277) | **Success Criteria**   * Use all five line sensors for proportional steering * Adapt to your environment with line calibration code | |
| **AP CSP Framework**  **DAT-2.E** Explain how programs can be used to gain insight and knowledge from data.  **AAP-1.C** Represent a list or string using a variable.  **AAP-2.H** Write conditional statements and determine their results.  **AAP-3.C** Develop procedural abstractions to manage complexity in a program by writing procedures.  **Computational Practice 2.B** Implement and apply an algorithm.  **Computational Practice 4.C** Identify and correct errors in algorithms and programs, including error discovery through testing. | **Materials**   * Several short tracks with lines for the ‘bots to follow. Each track should have a different curve. * Mission 6 Obj 7-8 Assignment / Answers * Solution code for LineFollow1\_final * [Mission 6 Kahoot Review](https://create.kahoot.it/share/firia-labs-codebot-mission-5/20d9499d-fe50-45a7-9f2c-623975832277) | |
| **Teacher Notes**   * Objective 7: EVen though CodeSpace gives a chart with possible data, it is still good for students to do this themselves. Each ‘bot needs a short track with a line. Follow the instructions in “Try Your Skills” to collect their own data. This doesn’t need to be recorded on their assignment, but you can ask students to create a chart. * Objective 7: Students already have a drive function. CodeTrek uses a SPEED\_LIMIT constant. Students can modify their drive() function to use it, but they don’t have to. * Objective 7: Give students time to test and modify their code as they try CodeBot on their course. If you have different courses, students can test their code on several and modify their speeds as needed. * Objective 8: There is A LOT of reading in CodeSpace. Students will be tempted to skip the reading and just answer the questions. But don’t let them! The instructions in CodeSpace really do tell the students what to do and why to do it. * Objective 8: Students will follow CodeTrek, but their code will look a little different because it is organized with functions at the top and the main program below. You can emphasize that this organization makes it easier to read, modify and understand their code. * Students again experiment with their code. Use the same courses as before. Students can adjust the speed in their code and try different courses to see what happens. The ‘bot should work really well in all situations. | | |