

Kodu Module 1: Eating Apples in the Kodu World

David S. Touretzky
Version of May 29, 2017

Learning Goals

- How to navigate through a world using the game controller.
- New idioms: “Pursue and Consume”, “Let Me Drive”
- Introduction to Kodu tile manipulatives.
- How to use the Kodu rule editor.
- First Law of Kodu: “See” always chooses the closest matching object.
- Couplets (recognizing pursue/consume pairs)

Worlds

- Treasure1, Treasure1b, Treasure2, Treasure2b
- Apple1X – with Debug LOS turned on for the kodu
- Star1X – with Debug LOS turned on for the kodu
- Star2 – with Debug LOS turned on for the kodu
- Flee1X – with Debug LOS turned on for the kodu
- FreeWorld1

Handouts

- Treasure Hunt
- Apple1X-Tips
- Star1X-Tips
- Closest1 World
- Flee1X World

Tile Manipulatives

- WHEN-see, WHEN-bumped
- DO-move, DO-eat
- apple (2 tiles), star (2 tiles)
- toward, it

Flash Cards

- Pursue and Consume
- Let Me Drive

Part 1: Using the Game Controller in a Treasure Hunt

1. This treasure hunt exercise is done in teams of two. Have students pair up.
2. Have students start Kodu and use the left stick to navigate to the “Load World” menu item. Note: if the cursor isn’t moving down the menu items to get to “Load World”, it could be because it’s stuck in the “Community News” window on the left. Move the left stick to the right to get to the main menu so you can scroll down to “Load World”.
3. Press the green A button to confirm.
4. Use the shoulder buttons (also called “bumpers”) to move to the “Downloads” tab. You should see a long list of worlds.
5. Press the yellow Y button to change the sort order, and use the left stick to select “Title”, then press the green A button to confirm. Now the worlds are sorted alphabetically by name.
6. Use the left stick to scroll through the worlds. Load and play the Treasure1 or Treasure1b world. (There are two versions so that students working in different teams can use different worlds and not use what they overhear from another team. The questions are the same for both worlds but the answers are different. Likewise, Treasure2 and Treasure2b will have the same questions but different answers.)
7. Hold up the game controller and teach the students these four controls:



8. Give students the Treasure Hunt worksheet and explain the Treasure Hunt task:
 - Student 1 is the “driver” who operates the game controller.
 - Student 2 is the “navigator” who reads and fills out the questionnaire.
 - For each object listed in the questionnaire, write down what object is next to it.



Part 2: Loading A Different Treasure World

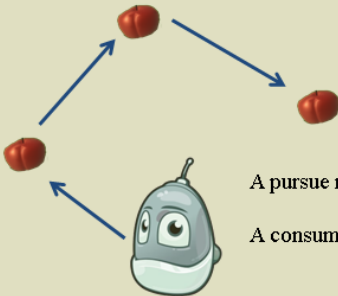
1. When the first treasure hunt is completed, load and play the Treasure2 or Treasure2b world.
2. Show students the Back button on the controller (see picture above).
3. To load the world, follow these steps:
 - a. Press the Start button to get to the Home Menu.
 - b. In the Home Menu, use the left stick to put the cursor on Load World.
 - c. Press the green A button to select Load World.
 - d. When the list of worlds is displayed, use the left or right shoulder buttons (also called “bumpers”) to move to the “Downloads” section. Don’t try to navigate through “All” or “Samples”: there are too many irrelevant worlds there.
 - e. If the worlds are not already sorted by title, press the yellow Y button and use the left stick to select “Title”, then press the green A button to sort.
 - f. Now use the left stick to scroll through the thumbnail images, listed alphabetically, to get to Treasure2 or Treasure2b. For students it might help to project your screen with your own cursor on the thumbnail so they can match visually instead of reading the names.
 - g. Press the green A button to select the world, then press A again to select “Play”.
4. With the new world loaded, have the students switch roles and complete the questionnaire for the second treasure hunt.

Part 3: The Apple1X World and the Pursue and Consume Flash Card

1. If working in pairs, have student 1 “drive” and student 2 navigate.
2. Load and play the Apple1X world.
3. Let the students move around and examine the world.
 - Point out the kodu and flying fish characters.
 - How many apples are there? (five)
4. Explain that the goal is to teach the Kodu to eat all the apples.
5. Give students the Pursue and Consume flash card and show them the rules for eating apples.

Pursue and Consume

Make the Kodu go to objects and eat them.



A pursue rule involves *motion*.

A consume rule *uses up* the object.

Pursue and Consume

| | | | | | | | | |
|---|------|--------|-------|---|----|------|--------|---|
| 1 | WHEN | see | apple | + | DO | move | toward | + |
| 2 | WHEN | bumped | apple | + | DO | eat | it | + |

Pursue rule

Consume rule

General Form:
 WHEN see *thing* DO move toward
 WHEN bumped *thing* DO *consume* it
 “Consume” can be “eat”, “grab”, “vanish”, or something else.

Filter by color:
 WHEN see *color thing* DO move toward
 WHEN bumped *color thing* DO *consume* it

Part 4: Introducing Tile Manipulatives

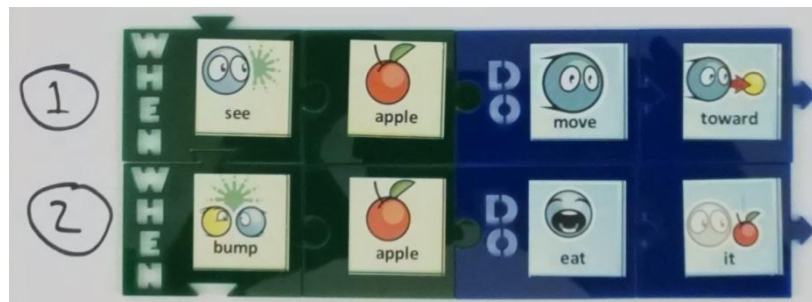
1. Distribute the tile manipulatives (one set for each pair of students).
2. Ask students to make the “pursue” rule from tiles; use the flash card for reference. Note: “star” or “ball” is on the back of the “apple” tiles, so students may need to turn these tiles over to find “apple”. The “it” is optional but makes the rule read better.

WHEN see apple DO move toward it

3. Ask students to make the “consume” rule from tiles, using the flash card for reference.

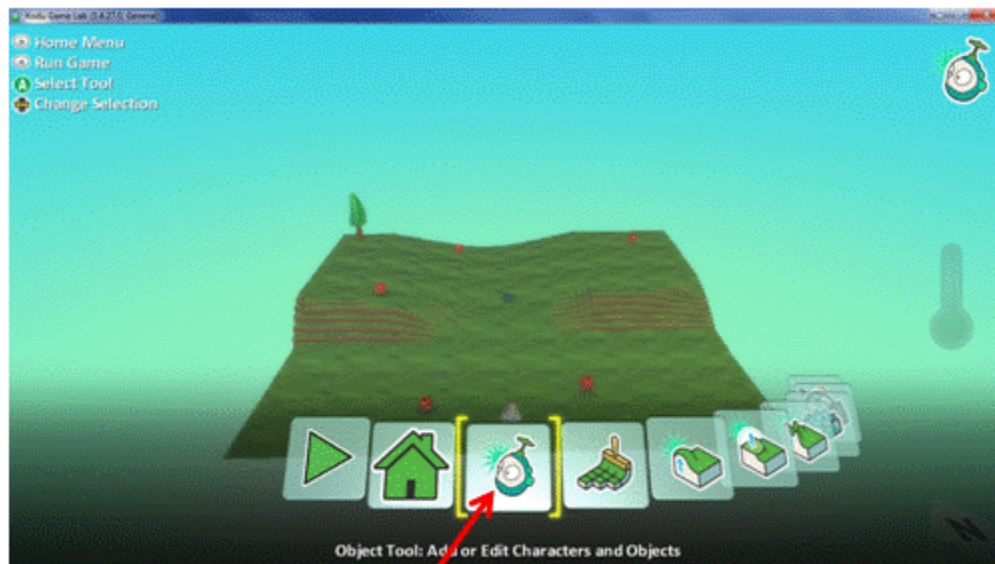
WHEN bumped apple DO eat it

4. Point out that the WHEN half of a rule is green and the DO half is blue.
5. Show the students how to link the two rows of tiles together.
6. Keep this tile structure handy as it will be used again in later steps of this session.

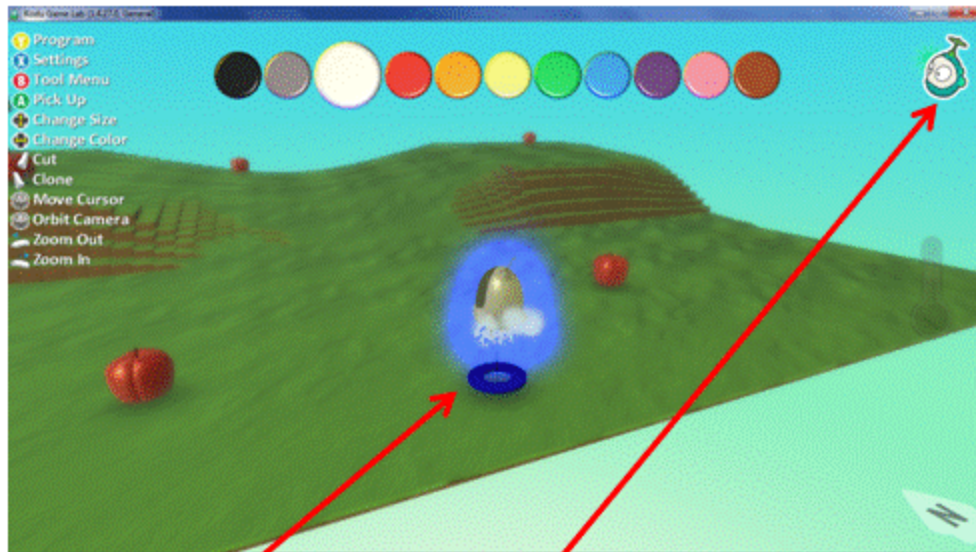


Part 5: The Kodu Rule Editor

1. Review the game controller buttons:
 - The Back button stops the game and brings up the tool menu.
 - Use the left stick to select the object tool (see images below) and the green A button to confirm.
2. Using the object tool, use the left stick to move the cursor (purple ring) under the kodu character. The kodu will glow blue when the cursor has selected it.
3. Press the yellow Y button to enter the rule editor and program the kodu.
4. Press the red B button to return to the object tool.
5. The only tools we'll use today are the object tool and the home page button. Tell students that the other tools can be ignored for now.
6. Point out that when the object tool (or any tool) is selected, help text appears in the upper left corner of the screen explaining what the various controller buttons will do when using that tool.



Object tool: press A to select.



Cursor from object tool is on the Kodu.
Press Y to program the Kodu.



Editing the Kodu's rules.

7. Press the Y button to enter the rule editor again. Guide students to enter the following rules. Note: words in parentheses indicate submenus, e.g. “apple” can be found in the “objects” submenu.

[1] WHEN see (*objects*) apple DO move toward it

[2] WHEN bumped (*objects*) apple DO eat it

8. A **common error** to watch for: selecting “move **forward**” instead of “move **toward**”.
9. If students make editing mistakes, show them how to fix things:
 - a. When the pencil is on the + sign, press the green A button to add a new tile.
 - b. When the pencil is on a tile:
 - The green A button changes the tile.
 - The blue X button deletes the tile.
 - c. When the pencil is on a rule number, the left trigger cuts (deletes) the entire rule.
 - d. Rule editor help is always visible in the top left corner of the screen.
10. Make sure students stay on page 1. If they accidentally move to another page, show them how to use the shoulder buttons to switch pages.
11. When finished editing, press the red B button (or the Back button) to leave the rule editor. Press the B button again (or use the Back button) to return to the tool menu.
12. **Running the program:** press the Back button. Depending on whether you’re at the tool menu, in the object tool, or in the rule editor, you may have to press the Back button once, twice, or three times.
13. **Stopping the program:** press the Back button once to stop the program and return to the tool menu.

Do Practice Sheet 1

Practice sheet 1 tests whether students can read and understand both the WHEN and DO parts of a rule. Students who do poorly on this sheet should be given additional practice to assure that they are actually reading the entire rule, not making assumptions about its contents based on past experience.

Part 6: Pursue and Consume

1. Review the Pursue and Consume flash card. What do these words mean?
 - “Pursue” means to chase, or to follow a path.
 - “Consume” means to eat, or to use up. (A car consumes gasoline; a fire consumes wood.)
2. What are some pursue actions in Kodu?
 - “move toward” and “turn toward”
 - Pursue actions always involve motion.
3. What are some consume actions in Kodu?
 - “eat it” is the only consume action we’ve seen so far. Another is “squash it”.
 - *There are other consume actions in the “combat” submenu, such as “boom” or “vanish”, but you might want to avoid this distraction for now.*
 - Consume actions remove an object, or change it so that it can’t be pursued any more.

➤ “Squash” is an example of the latter; squashed objects are no longer seen. Another example, too advanced to discuss right now, is to pursue objects of a particular color and consume them by changing their color, as in the Paint1 world.

4. Some actions are neither pursue nor consume actions.
 - “play” plays a sound. It doesn’t involve motion and it doesn’t use up an object.
5. The WHEN part of a rule doesn’t determine whether it’s a pursue or consume rule.
 - Not every “WHEN see” rule is a pursue rule; the DO part must say “move toward” or “turn toward”.
 - Not every “WHEN bumped” rule is a consume rule; the DO Part must say “eat it” or “squash it” or some other consume action.

Do Practice Sheet 2

Practice sheet 2 tests whether students understand the concepts “pursue” and “consume”. Having them label rules with a “P”, “C”, or “X” sets the stage for introducing “couplets” later.

Part 7: Which Object Will the Kodu Pursue?

1. Each time we run the Apple1X world, the kodu does the same thing. How does it decide which apple to pursue first? And after it eats the first apple, how does it decide which apple to pursue next?
2. Let the class discuss this and come to a conclusion.
3. Explain that the kodu behaves this way because of the First Law of Kodu: each rule picks the closest matching object. Show them the graphic or hand out the fridge magnet.



4. In the graphic, which apple is the Kodu pursuing? (The closest one.)

5. How can we test the First Law? We can change the kodu's starting position.
 - a. If the program is running, press the Back button to stop it.
 - b. Select the object tool.
 - c. Put the cursor on the kodu.
 - d. Press the green A button to pick up the kodu. The cursor turns orange.
 - e. Use the left stick to move the kodu to a different starting position.
 - f. Press the green A button to put the kodu down. The cursor turns purple again.
 - g. Press the Back button to run the program.
6. Show the YouTube video "The First Law of Kodu". (Google the title to find it.)

Do Practice Sheet 3

Practice sheet 3 tests whether students can apply the First Law. A common error is to assume that the kodu cannot see behind it; students need to understand that the kodu sees in all directions. Another common error is to think the kodu will not see an object that is behind another one. The kodu sees everything, and it will push its way past any object that is in the way of the object it's trying to reach.

Part 8: Why Do We Need A Consume Rule?

1. Have students go back into the rule editor and delete the consume rule, leaving only the pursue rule. To do this, put the pencil on the rule number "2" and press the left trigger to delete the rule. Now the page looks like this:

[1] WHEN see apple DO move toward

2. Ask students what they think will happen when we run the program.
3. Answer: the kodu goes to the closest apple (First Law) and gets stuck there. It never visits any of the other apples because it never consumes the first one.
4. Run the program and observe that nothing happens

Part 9: Why Do We Need A Pursue Rule?

1. Have students restore the consume rule and delete the pursue rule. The page will look like this:

[1] WHEN bumped apple DO eat it

2. Ask them to predict what will happen when they run this program.
5. Answer: nothing happens. The kodu isn't eating any apples because it hasn't bumped any. None are in reach, and it's not moving. We have consume without pursue.

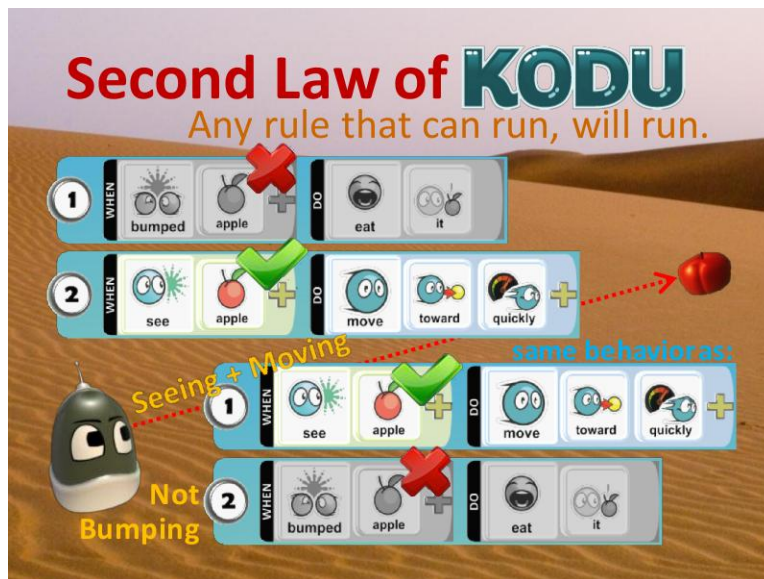
- Use this opportunity to emphasize that “bumped” is a predicate, not an action. The consume rule does not tell the kodu to go off and bump an apple; it tells the kodu how to react when a bump occurs. If nothing is moving, there will be no bumping. We need a pursue rule to make the kodu move.

Part 10: Consume Followed by Pursue

- Have students re-enter the pursue rule so that it’s now the second rule. If you need to move a rule, put the pencil on the rule number, press the green A button to pick it up, use the left stick to move the rule up or down, and press “A” again to put the rule down.
- Make sure students don’t accidentally indent a rule when moving it up or down.
- Now we have Consume and Pursue instead of Pursue and Consume. The page should look like this:

[1] WHEN bumped apple DO eat it
 [2] WHEN see apple DO move toward

- Ask students what the kodu will do given a Consume and Pursue program.
- Have them run it and see.
- Answer: rule order doesn’t matter in this case; it will behave the same as when the order was Pursue and Consume. *Any rule that can run, will run.* This is the Second Law of Kodu.



- Show the Second Law graphic or hand out the fridge magnet. Explain what’s going on in the graphic:
 - There are two programs shown. The top one is “consume and pursue”, while the bottom one is “pursue and consume”, but the rules are the same, and so is the result.
 - The kodu isn’t bumping anything, so the consume rule can’t run (red “X”).
 - The kodu does see an apple, so the pursue rule can run (green check mark).
 - Both programs behave the same way. The rule order doesn’t matter here.

8. Put the table below on the blackboard and ask students to help fill out the second column:

| Program | What Happens |
|--------------------|---------------------------|
| Pursue and Consume | Eats all the apples |
| Pursue alone | Gets stuck at first apple |
| Consume alone | Can't move |
| Consume and Pursue | Eats all the apples |

Show the first part of the Rule 2 video to reinforce the Second Law.

Part 11: Bringing Apples to the Kodu with “Let Me Drive”

1. Have students read the Let Me Drive flash card.
2. Tell them to program the flying fish with the first Let Me Drive rule so they can drive it with the left stick. They don't need any other rules.

[1] WHEN gamepad L-stick DO move

3. Delete the kodu's pursue rule, leaving just its consume rule.
4. When running the program, use the left stick to make the flying fish push apples to the kodu. When an apple bumps the kodu, the kodu will eat it.
5. Conclusion: the consume rule still works, as long as apples somehow reach the kodu.
6. Question for the students: what happens if the flying fish bumps the kodu? Answer: nothing, because the consume rule is configured to only eat apples.

Part 12: Competing with the Kodu (Optional)

1. Have students go back into the rule editor and re-enter the kodu's pursue rule.
2. Run the program. Now their goal is to use the flying fish to keep the kodu from eating all 5 apples before the time runs out. In addition, the flying fish can score 2 points for every apple it drops into the pit where the kodu can't get it. If the flying fish ends up with at least 4 points, it wins.

Part 13: Programming Another World: Star1X

1. Remind students of the four motion controls: left stick moves the cursor, right stick orbits the camera around the cursor, and the left and right shoulder buttons zoom out and in, respectively, centered on the cursor.
2. Give them the Star1X handout.
3. Have them load and play the Star1X world. During the Load World dialog, press the blue X button to discard the changes made to the Apple1X world.
4. Have the students explore the Star1X world.
5. Explain that the goal is to have the Kodu eat all the stars. The octopus moves stars around randomly.
6. Optional: have the students use the tile manipulatives to represent the program. Note: they should recognize that the rules will be the same as for the Apple1X world, except “apple” is replaced by “star”. Two of the four “apple” tiles have “star” on the back.



7. Have the students program the kodu to eat the stars. Note: to find the star tile in the editor they must select “more” from the objects menu.

[1] WHEN see (*objects*) (*more*) star DO move toward

[2] WHEN bumped (*objects*) (*more*) star DO eat it

Part 14: Reasoning About Behavior in Star1X: Closest Matching Object

1. Ask the students: *How does the kodu decide which star to eat first?* Answer: it always goes to the closest star.
2. Ask the students: *Which law accounts for this?* Answer: the First Law: each rule chooses the closest matching object.
3. Have them complete the second page of the Star1X activity. This will help the students test their hypothesis by moving the kodu to new starting places.
4. Students might decide on their own to move the stars instead of or in addition to moving the kodu. That’s okay; they can still investigate the question of where the kodu decides to go.
5. Ask the students: *Can the kodu ever get stuck? When does that happen?* Answer: the kodu can get stuck if there is a tree between it and the closest star.
6. Ask the students: *Does the octopus help or interfere with the kodu?* Let the class discuss this. Answer: sometimes the octopus interferes by grabbing a star the kodu was trying to get. But if the

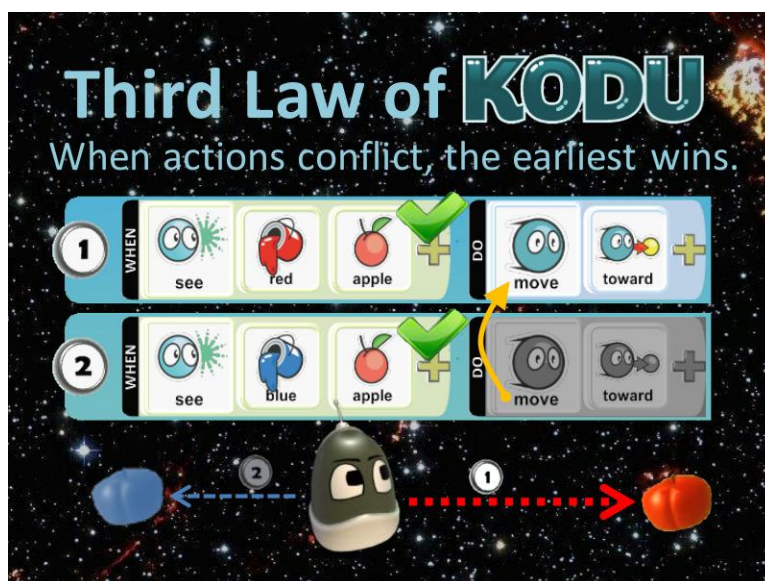
kodu gets stuck, the only way it can get unstuck is for the octopus to move one of the stars so that the kodu changes direction.

Part 15: The First Law of Kodu and the Closest1 World

1. Ask students to recite the First law of Kodu. It is: **“Each rule picks the closest matching object.”** We call this the “Closest Law”.
2. Test their understanding by having them load the Closest1 world and complete the Closest1 exercise. For young students this can be done as a class participation exercise, where students call out what they think is the right answer and the teacher then runs the rule to check. Older students can do the exercise on their own, or as a homework assignment.

Part 16: Eating Two Things: Star2 World and the Third Law of Kodu

1. Distribute the Star2 handout.
2. Have students load the Star2 world. Notice that it contains hearts and coins.
3. Ask the students to program the kodu to eat both the hearts and the coins. Their likely solution:
 - [1] WHEN see heart DO move toward
 - [2] WHEN bumped heart DO eat it
 - [3] WHEN see coin DO move toward
 - [4] WHEN bumped coin DO eat it
4. Run the program and have students fill out the table on page 2 of the Star2 handout showing the order in which objects are eaten. The kodu eats all the hearts before eating any coins. Ask the students why they think it does this.
5. Answer: this behavior is a consequence of the Third Law of Kodu, the “Conflict Law”: *When actions conflict, the earliest wins.* The fridge magnet shows that the kodu will go to the red apple even though the blue one is closer, because rule 1 always wins over rule 2 if rule 1 can run. If there are no red apples, rule 1 cannot run, so rule 2 will be able to run.



Part 17: Using Couplets to Analyze Pursue and Consume Programs

Teach students to use couplets using Star2 and the first two pages of the 01-Couplets handout.

Do Practice Sheet 4

Part 18: Additional Practice With Multiple Pursue and Consume Rules

Return to the Star 2 handout and have students try the exercises on pages 3-4.

1. **Rule Scramble 1:** Have students swap the first two rules, so that the consume hearts rule is the first rule and the pursue hearts rule is the second. To do this in the rule editor:
 - a. Put the pencil on the rule number of the pursue hearts rule (rule 1).
 - b. Press A to pick up the rule.
 - c. Use the left stick to move the rule down, to the second position. Make sure you do not accidentally indent the rule.
 - d. Press A again to drop the rule.
2. Ask students if they think this change will make a difference. Have them run the program and fill out the table with their observations. (Answer: the switch makes no difference.)
3. **Rule Scramble 2:** Have students move the pursue coin rule *before* the pursue hearts rule.
4. Run the program again and observe that now the kodu eats all the coins first, then the hearts.
5. **Rule Scramble 3:** Would it make a difference if we swapped the order of the two consume rules?
6. Answer: no difference, because the consume rule only runs when the kodu bumps an object, and what gets bumped next is determined by the pursue rule.

Part 19: Solving Tricky Cases Using Couplets

1. Return to the 01-Couplets handout and cover the tricky cases described there.
2. Exercise: starting with the Star2 world and its four rules, have a student pick one rule to delete.
3. Apply the couplets method to analyze the resulting three-rule program and predict the effect.
 - a. What gets eaten?
 - b. What gets ignored?
 - c. Will the kodu get stuck?
4. Run the program to test your predictions.
5. Repeat the exercise by restoring the deleted rule and deleting a different one.

Do Practice Sheet 5

Part 20: Flee is the Opposite of Pursue (Advanced Exercise)

1. Distribute the Flee1X World handout.
2. Let students read the instructions and try to complete the exercise.
3. By looking at the tile menu, students should recognize that they can use “move away” to flee from the pushpad.
4. Note: “move avoid” will sort of work, but not very well; the kodu flees more aggressively with “move away”. If a student chooses “move avoid”, suggest they look for another tile to use.

5. The “move away” rule must come *before* the rule to move toward coins.
6. They can also use Let Me Drive to drive the octopus, and use it to push coins toward the kodu, or push the kodu away from the pushpad. But if the octopus bumps a pushpad it will be knocked out.
7. Let students write the code on their own.

Do Worksheet 3

Part 21: Create You Own Pursue and Consume Example

1. Have students load the FreeWorld1 world.
2. Show them how to add objects to the world using the object tool.
3. Show them how to change the color of an object using the D-Pad.
4. Invite them to populate the world with objects of their choosing.
5. Then, have them program the kodu to demonstrate some form of Pursue and Consume behavior.
6. Ideas to consider:
 - a. Have the kodu pursue and consume all the trees.
 - b. Put in multiple kodus of different colors and have them eat each other; see who is left.
 - c. Program the kodu to pursue and consume apples; program a cycle to pursue and consume kodus.

Do Worksheet 4