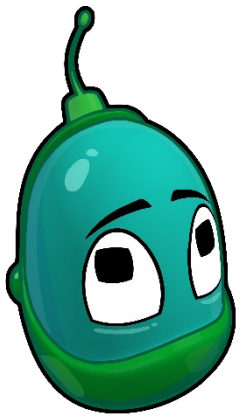
Teaching “Lawfulness” With Kodu

David S. Touretzky
Carnegie Mellon University

Christina Gardner-McCune
Ashish Aggarwal
University of Florida



SIGCSE '16, Memphis TN

Funded in part by a gift from Microsoft Research.

Gravity.

It's not just a good idea.
It's the Law.

Essence of Computational Thinking:

The lawful manipulation of structured representations.

Evidence for Mastery of Lawfulness

Children should be able to:

1. **State** the laws.

2. **Explain** program behavior in terms of the laws.

3. Use the laws to **predict** future behavior from current state. This involves mental simulation.

Aspects of Lawfulness in Kodu

1. Syntactic structure of Kodu programs

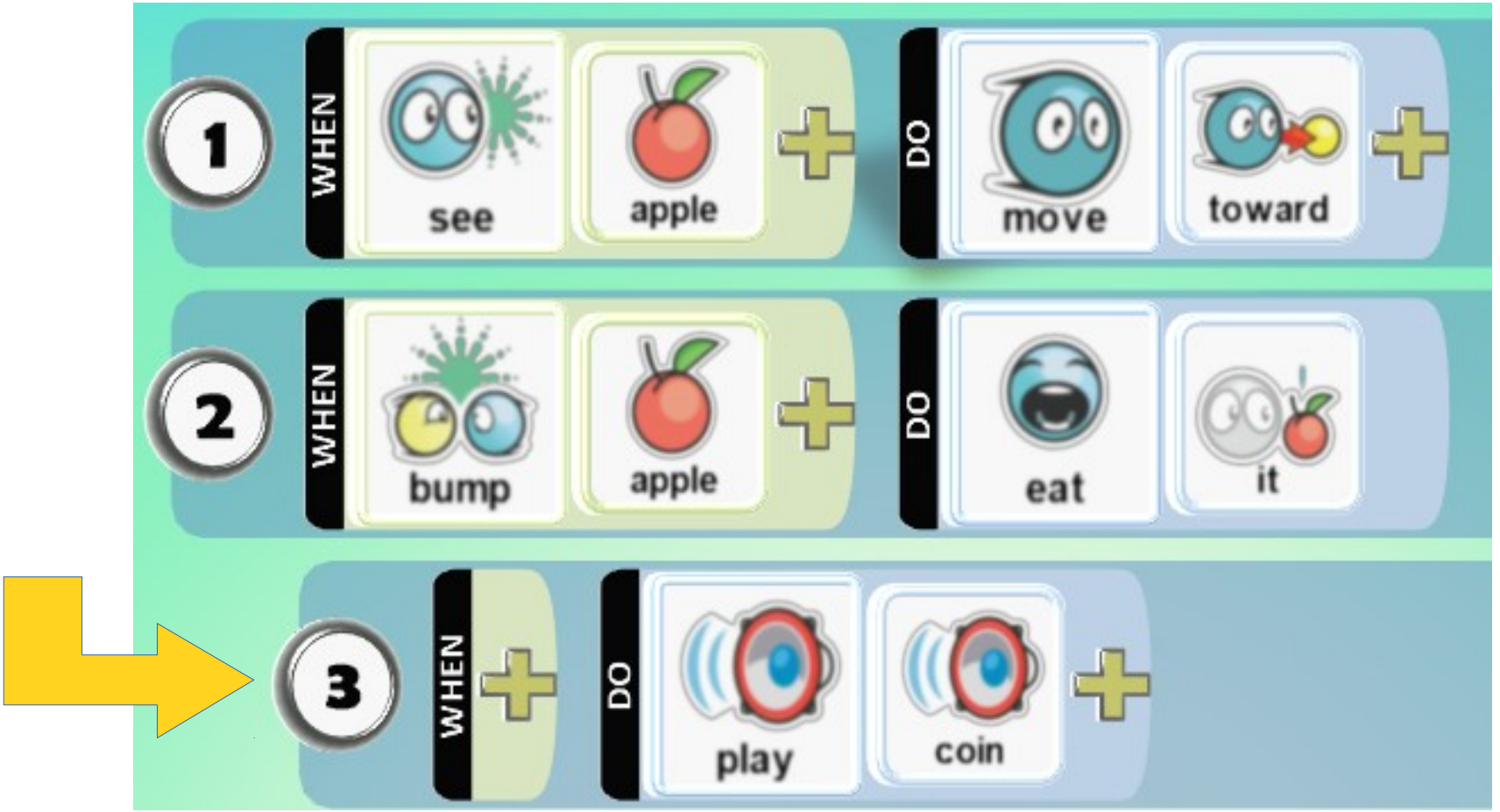
2. Kodu design patterns (idioms)

3. Principles of Kodu computation (semantics)

4. State machine formalism

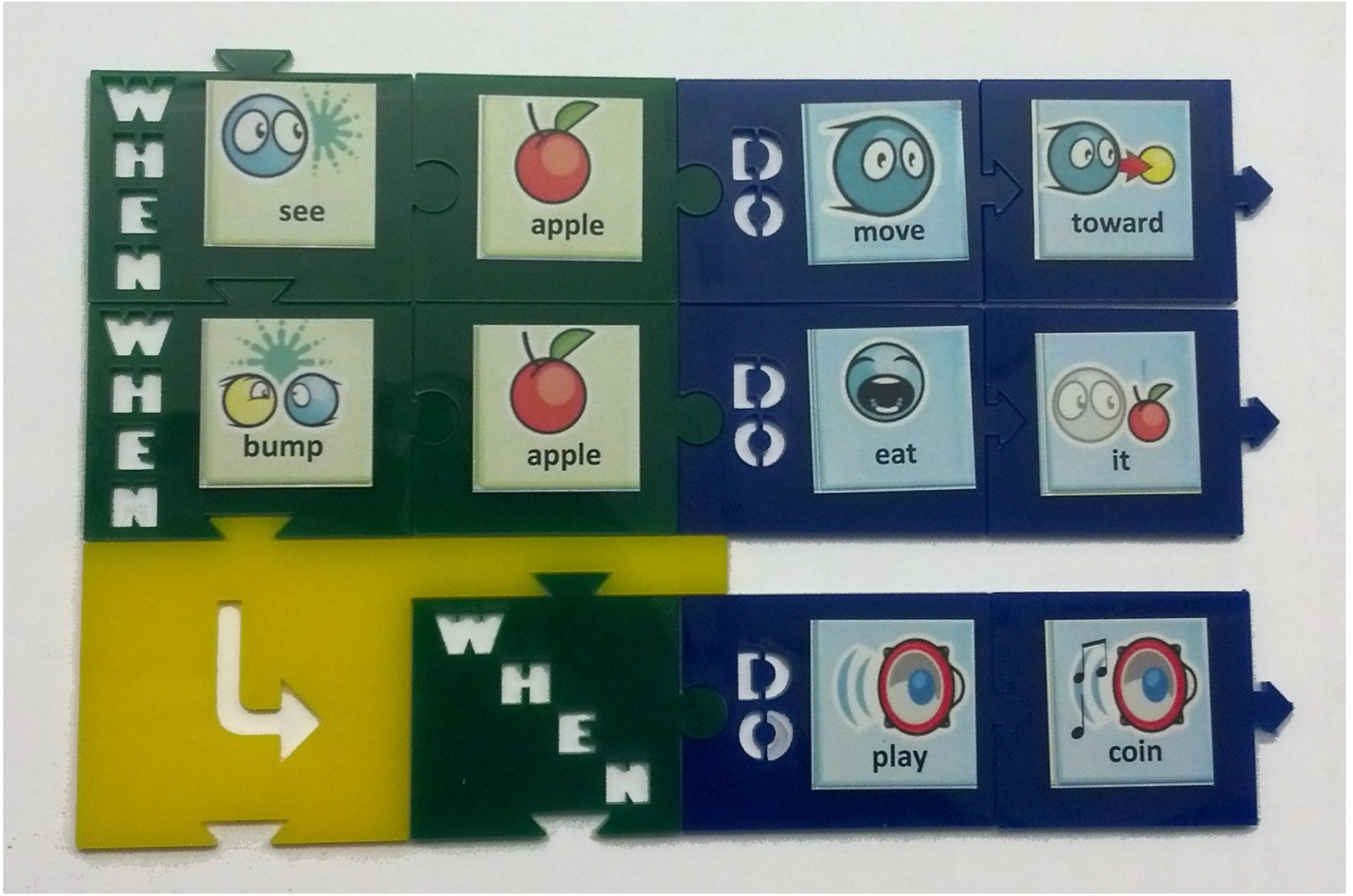
Our curriculum provides scaffolding for lawfulness.

1. Syntactic Structure



Rules have a WHEN phrase and a DO phrase.
Each phrase begins with a predicate (for WHEN) or action (for DO).
Nouns appear in the WHEN phrase; pronouns (“it” or “me”) in the DO phrase.
Indentation denotes rule dependency and block structure.

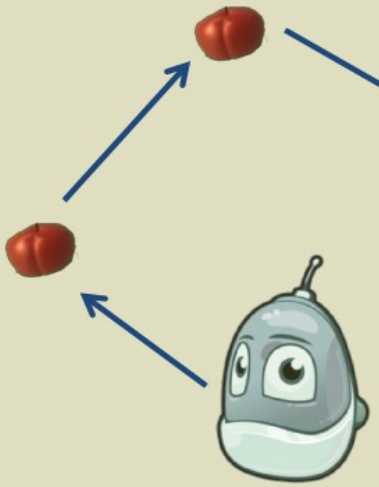
Tile Manipulatives



2. Kodu Idiom Flash Cards

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

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

Kodu Idiom Flash Cards

Do Two Things

Make the Kodu take two actions with one rule.

WHEN *something* … DO **this**

and also → DO **that**

Do Two Things

When you've bumped an apple, eat it *and also* play the coin sound.

General Form:

WHEN *something* DO *action1*

↳WHEN DO *action2*

Indenting the second rule makes it dependent on the WHEN part of the rule above.

Kodu Idiom Catalog

- **Pursue and Consume**
- **Do Two Things**
- Count Actions
- Default Value
- Show Page As Color
- Follow the Yellow Brick Road
- Random Choice
- Let Me Drive
- Visible Stopwatch
- Countdown Timer
- Once Is Enough
- Parting Shot
- If This And Also That

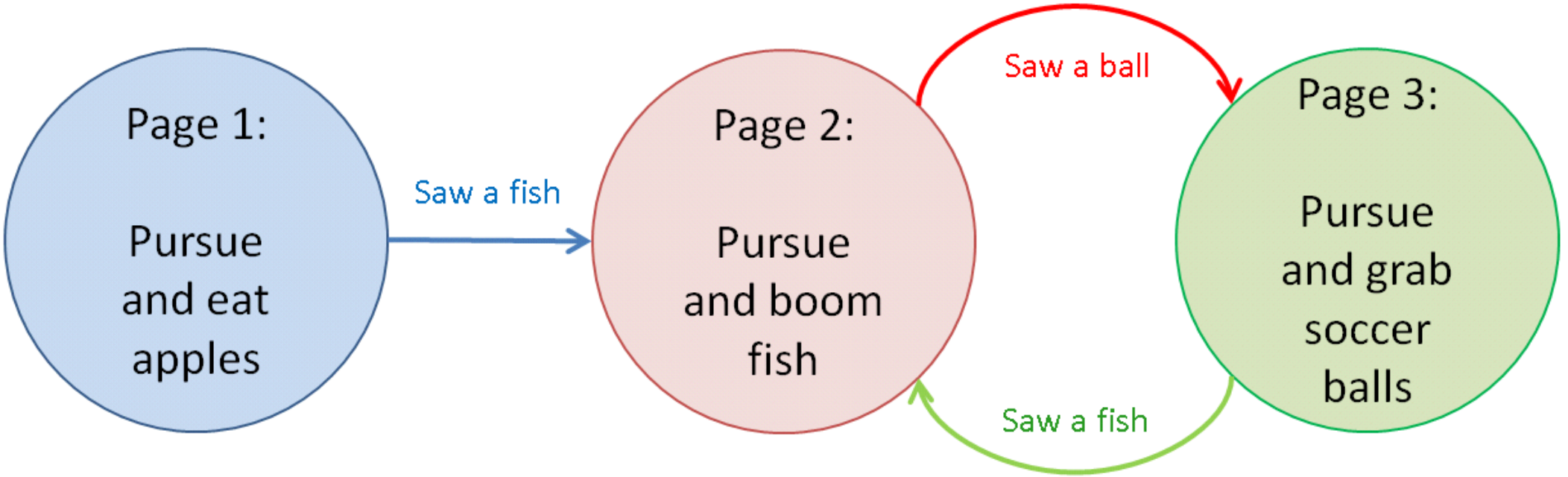
3. Principles of Kodu Computation

- Rules pick the closest matching object.
- Filters work together to constrain the match.
- An indented rule can run only if its parent's WHEN part is true.
- When actions conflict, the lower numbered rule wins.

Above are the basic principles; there are many more.

Study these: a quiz is coming up!

4. State Machine Formalism



PAGE 1:

[1] WHEN see apple DO move toward
[2] WHEN bumped apple DO eat it
[3] WHEN see fish DO **switch to page 2**

After grabbing a soccer ball, can the kodu ever eat another apple?

Our Study

- Two separate week-long summer camps:
  Monday to Friday, 3 hours/day

- 23 participants: rising 5th and 6th graders
  - Generally high SES families
  - 26% female (6 female, 17 male)
  - 14 White
    4 Asian/Indian
    1 Latino
    1 Multiracial
    1 Native American

Prior Experience

- 4 had no prior programming experience.
- 12 had participated in 2 or more computing programs; 5 had done 5+ computing programs.
- Prior activities included:
  - Scratch (12)
  - Minecraft (9)
  - Hour of Code (9)
  - Robotics (5)
  - Python (7)
  - HTML and Javascript (4)
  - Kodu (1)

Assessing Mastery

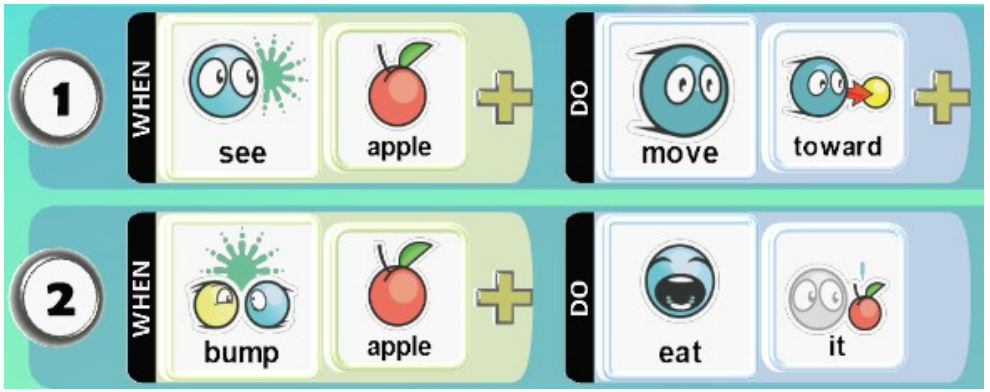
Children who have mastered “lawfulness” should be able to:

1. **State** the laws

2. **Explain** program behavior in terms of the laws.

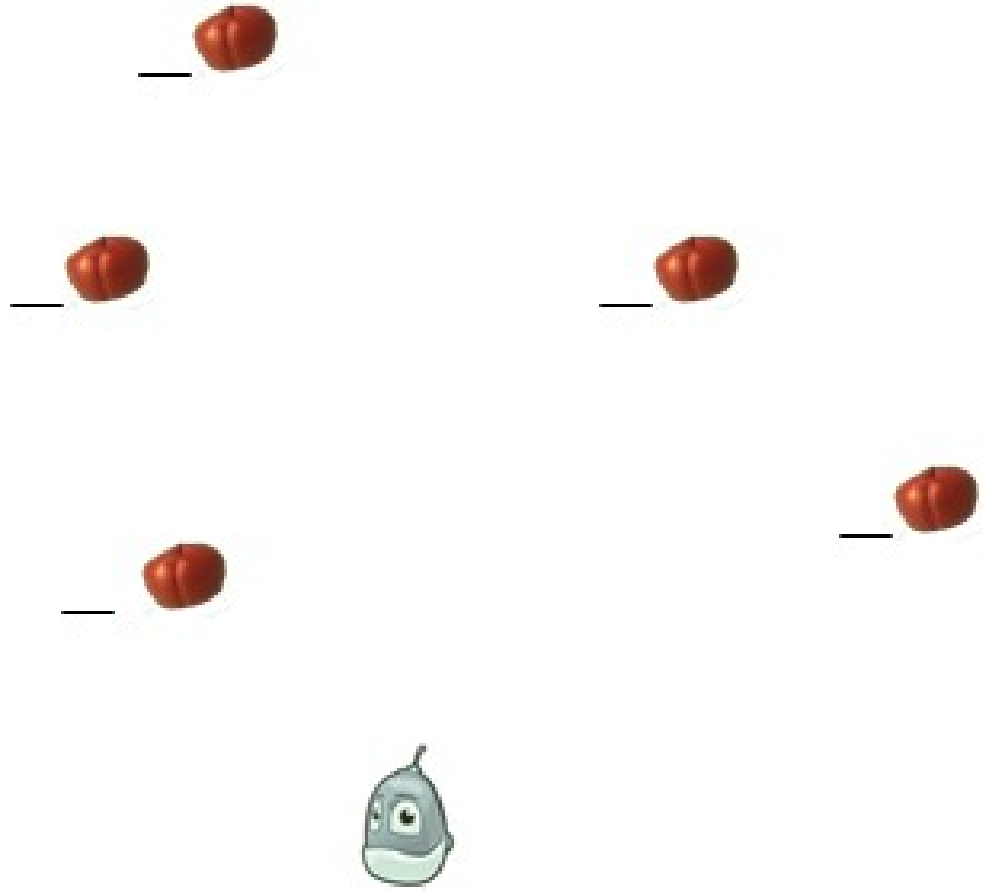
3. Use the laws to **predict** future behavior from current state. This involves **mental simulation**.

Day 1 Mental Simulation

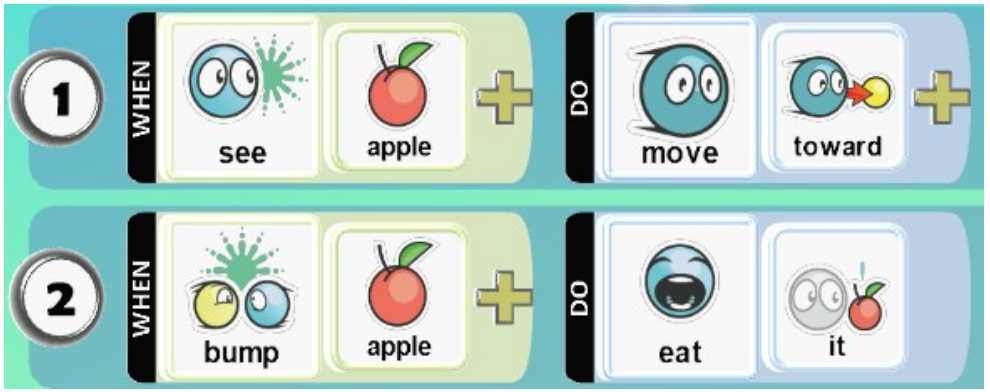


Idiom: Pursue and Consume

Principle: closest matching object.

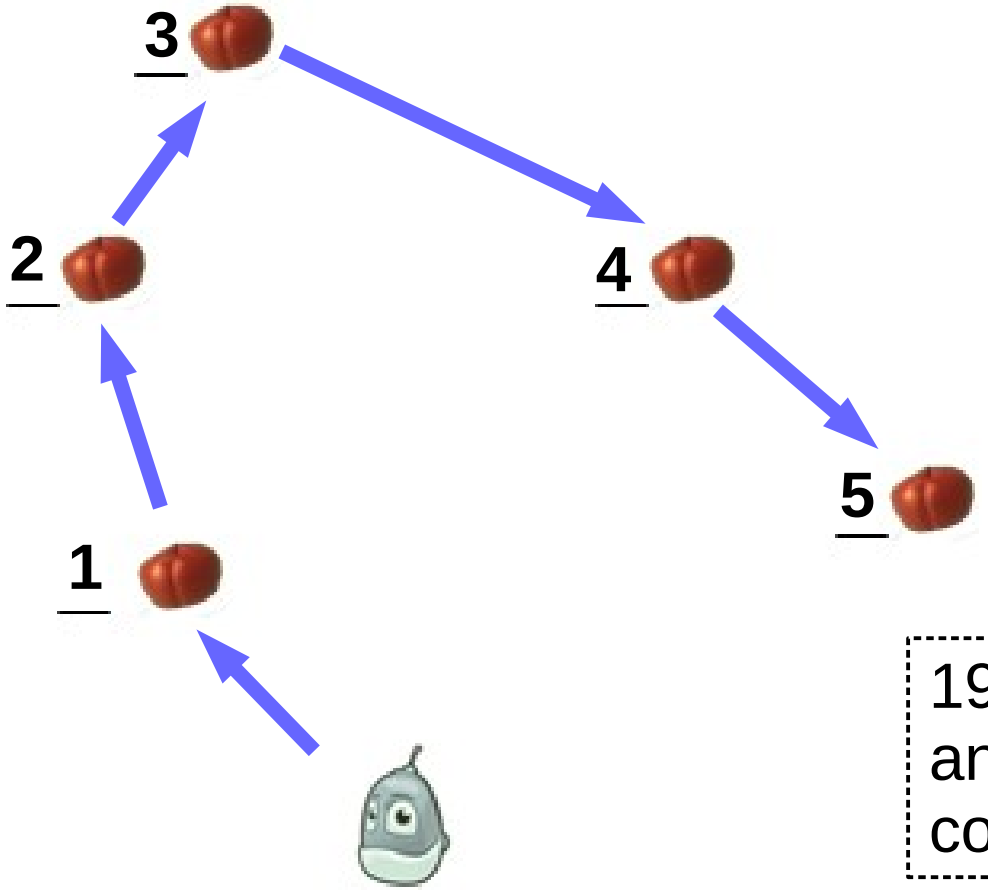


Day 1 Mental Simulation (Correct)



Idiom: Pursue and Consume

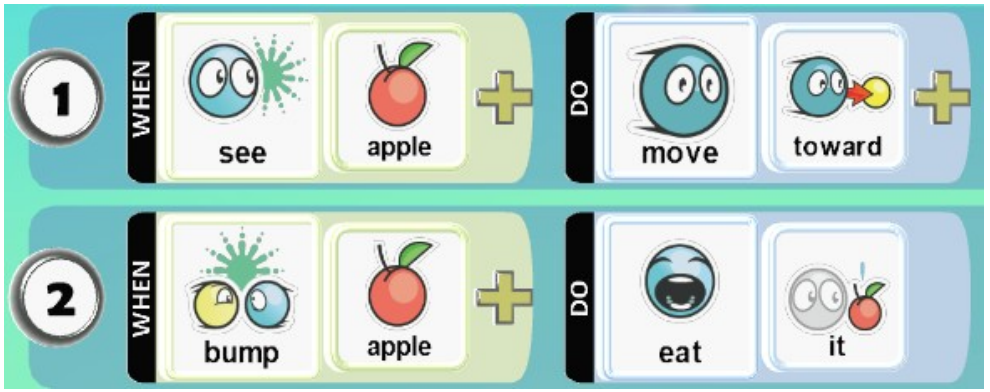
Principle: closest matching object.



Day 1

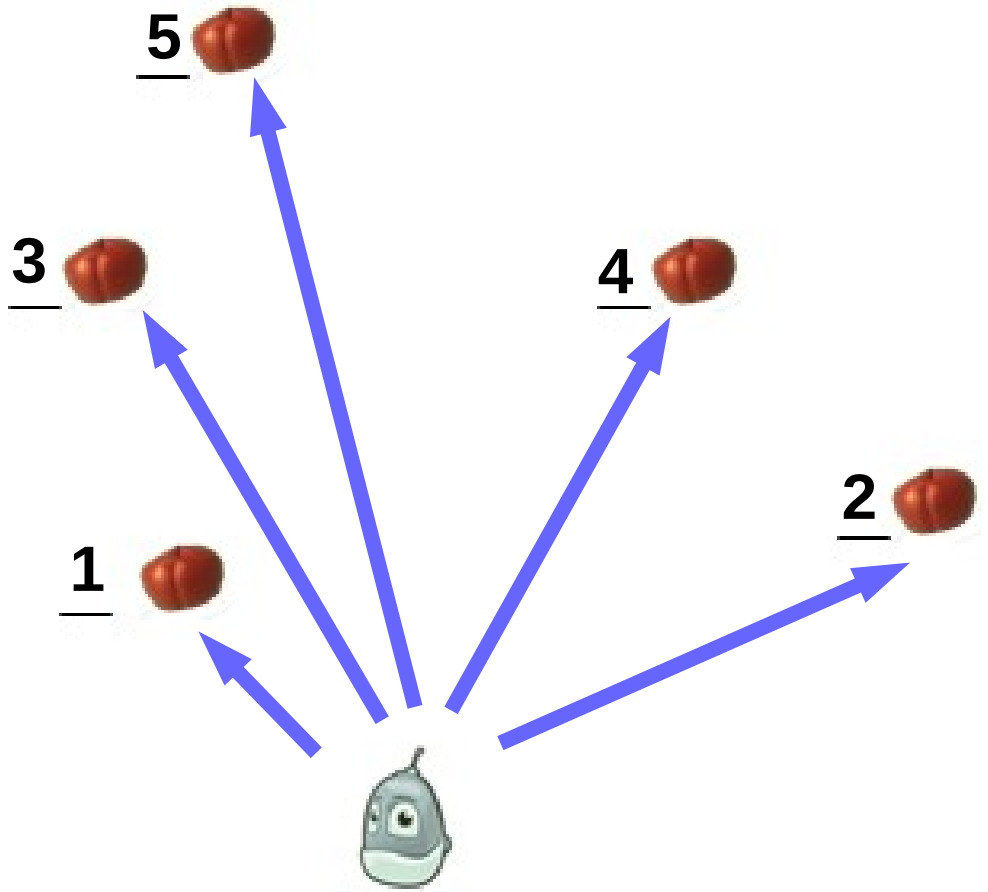
19/23 (91%) answered correctly

Day 1 Mental Simulation (Faulty)

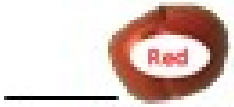


Idiom: Pursue and Consume

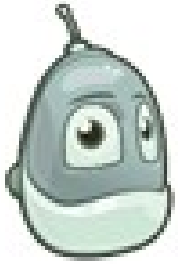
Principle: closest matching object.

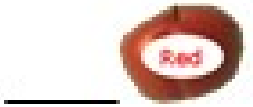


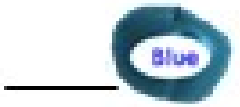
Day 4 Q2

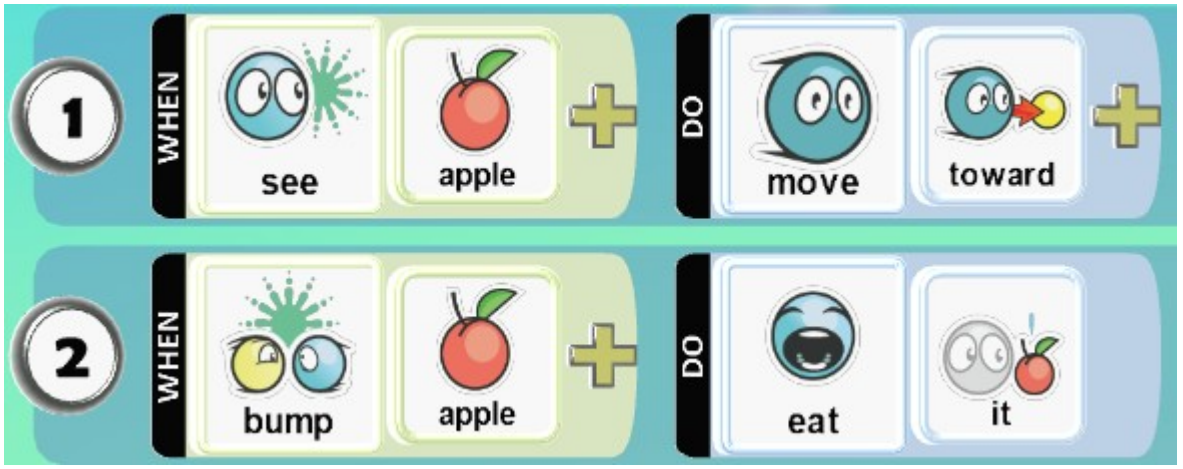

Red

Blue

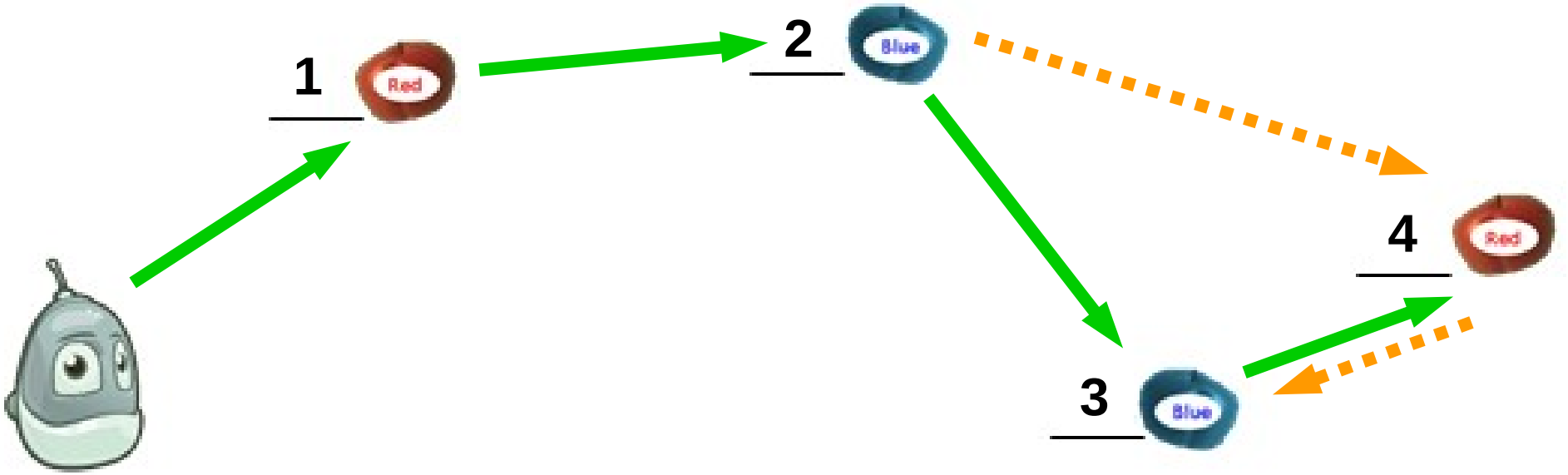


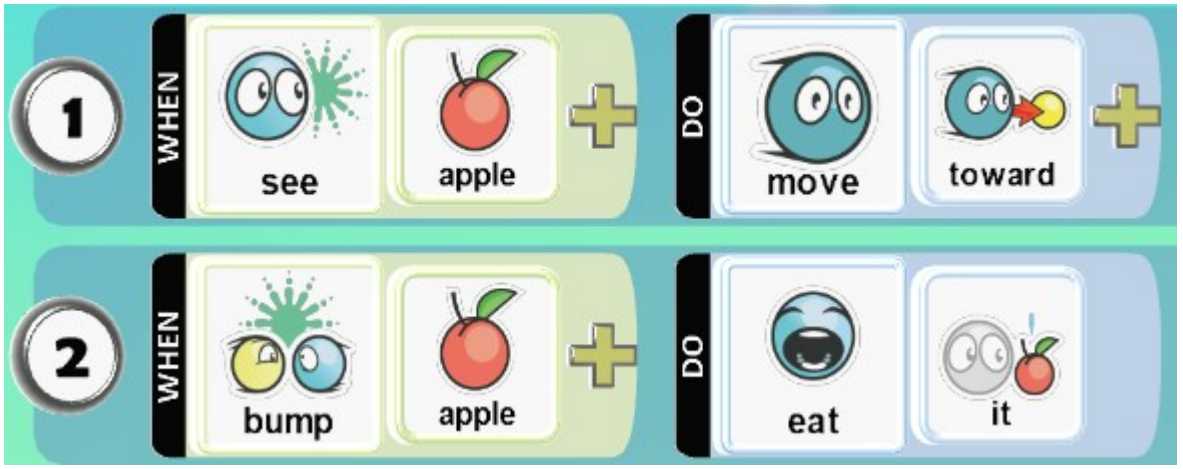

Red


Blue


1
WHEN
see
apple
DO
move
toward
2
WHEN
bump
apple
DO
eat
it

Day 4 Q2





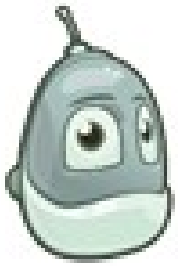
18/23 (78%) answered correctly: **1-2-3-4**.

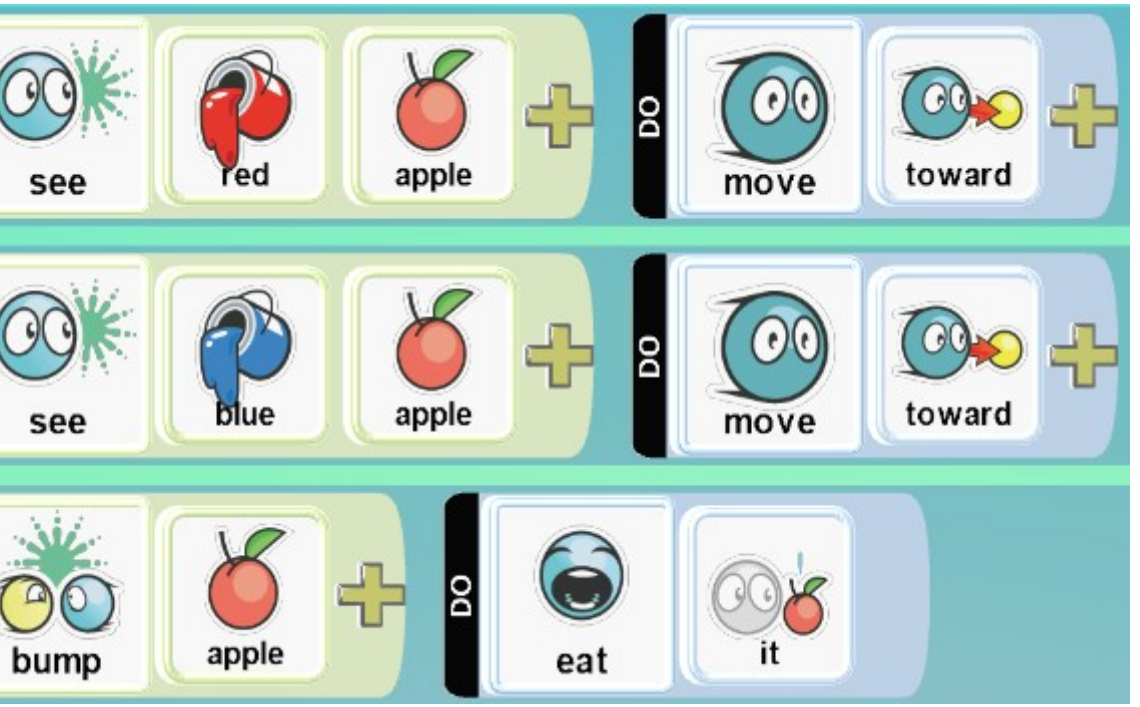
3/23 answered **1-2-4-3**. Did they mis-perceive "closest"?

Understanding Rule Ordering

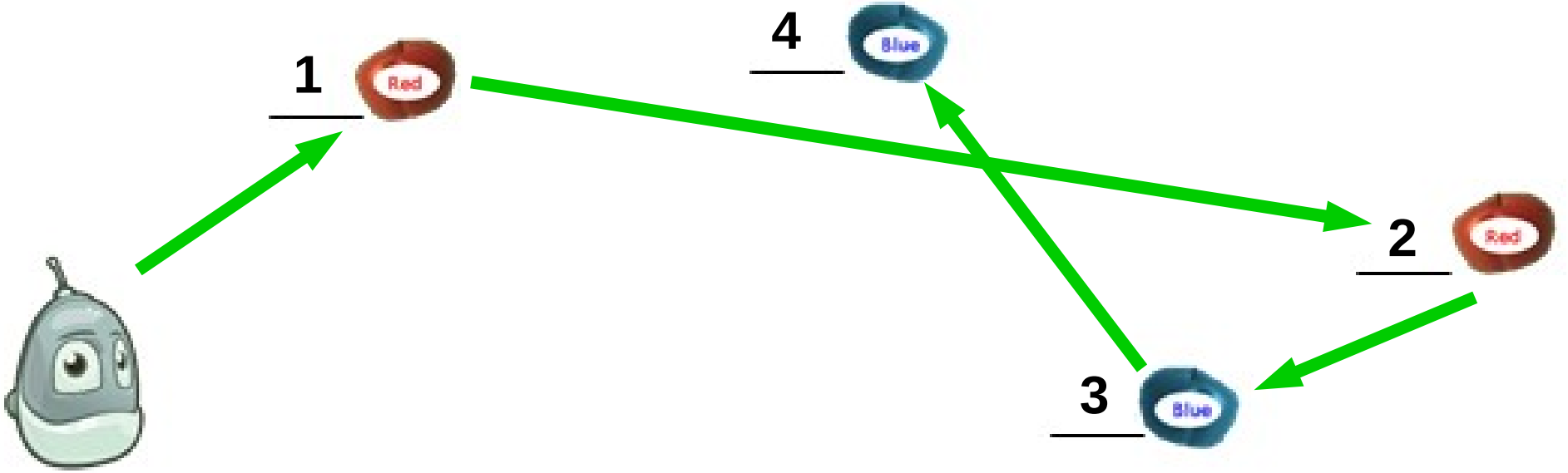
- In general, rule ordering doesn't matter.
- But when actions conflict, the lower numbered rule wins (fourth principle).

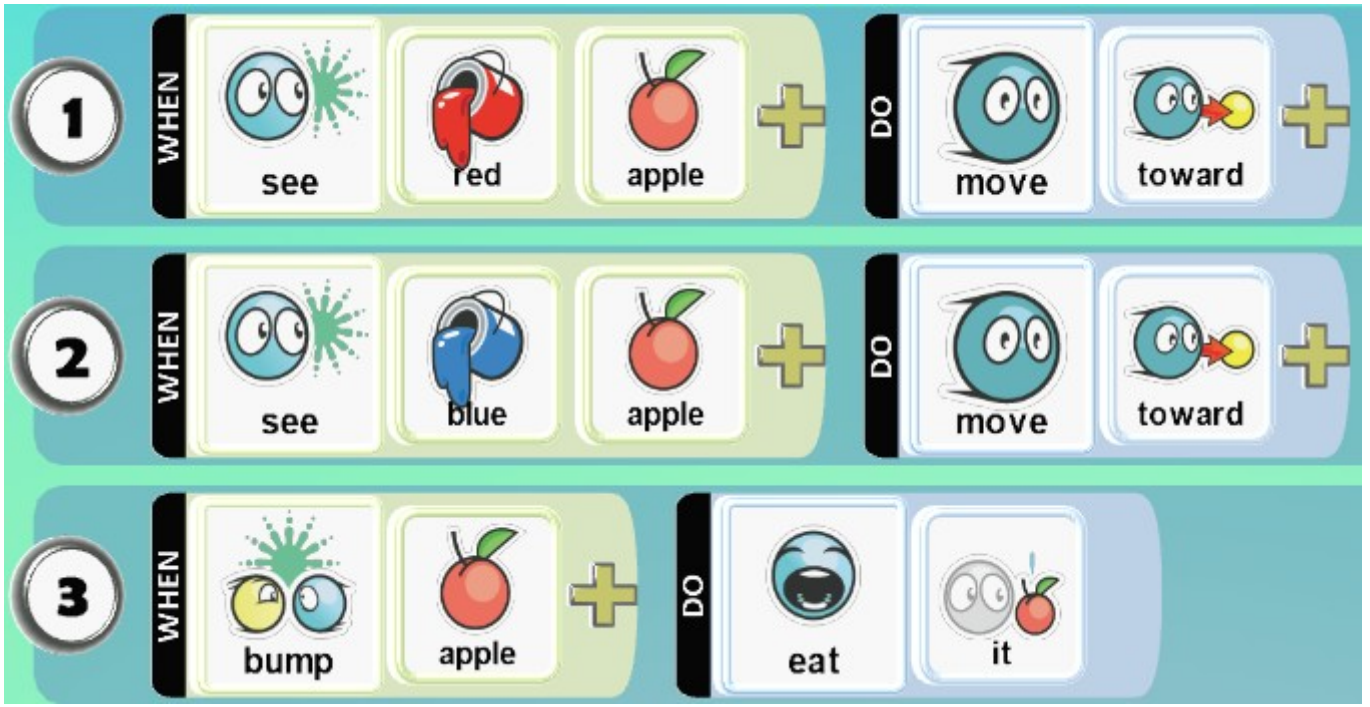
Day 4 Q3




1
WHEN
see
red
apple
DO
move
toward
2
WHEN
see
blue
apple
DO
move
toward
3
WHEN
bump
apple
DO
eat
it

Day 4 Q3



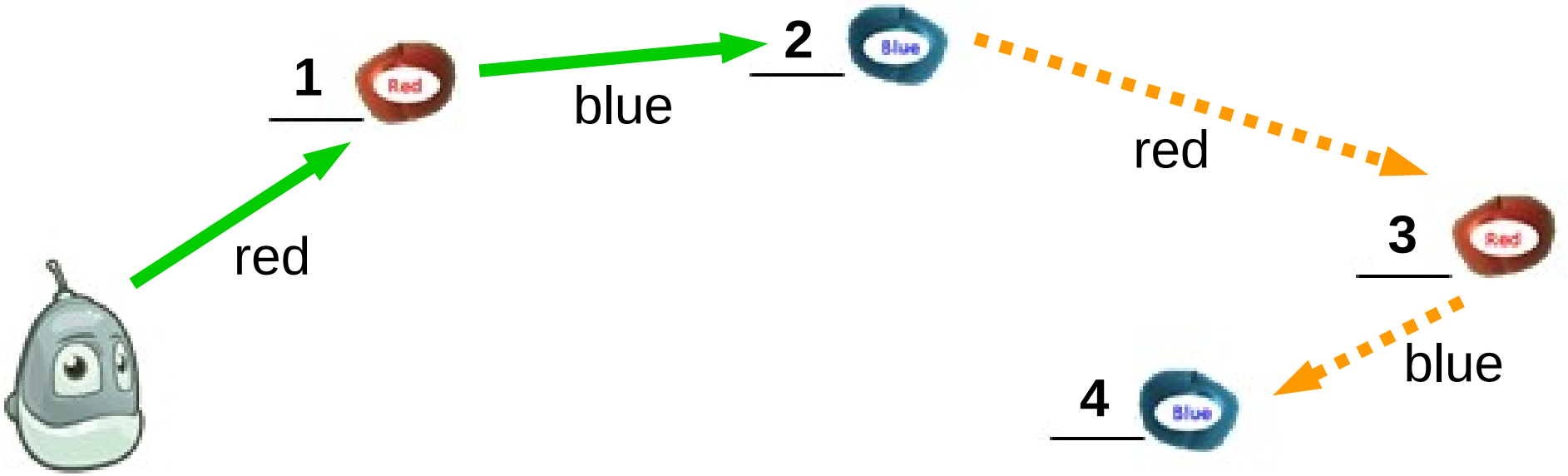


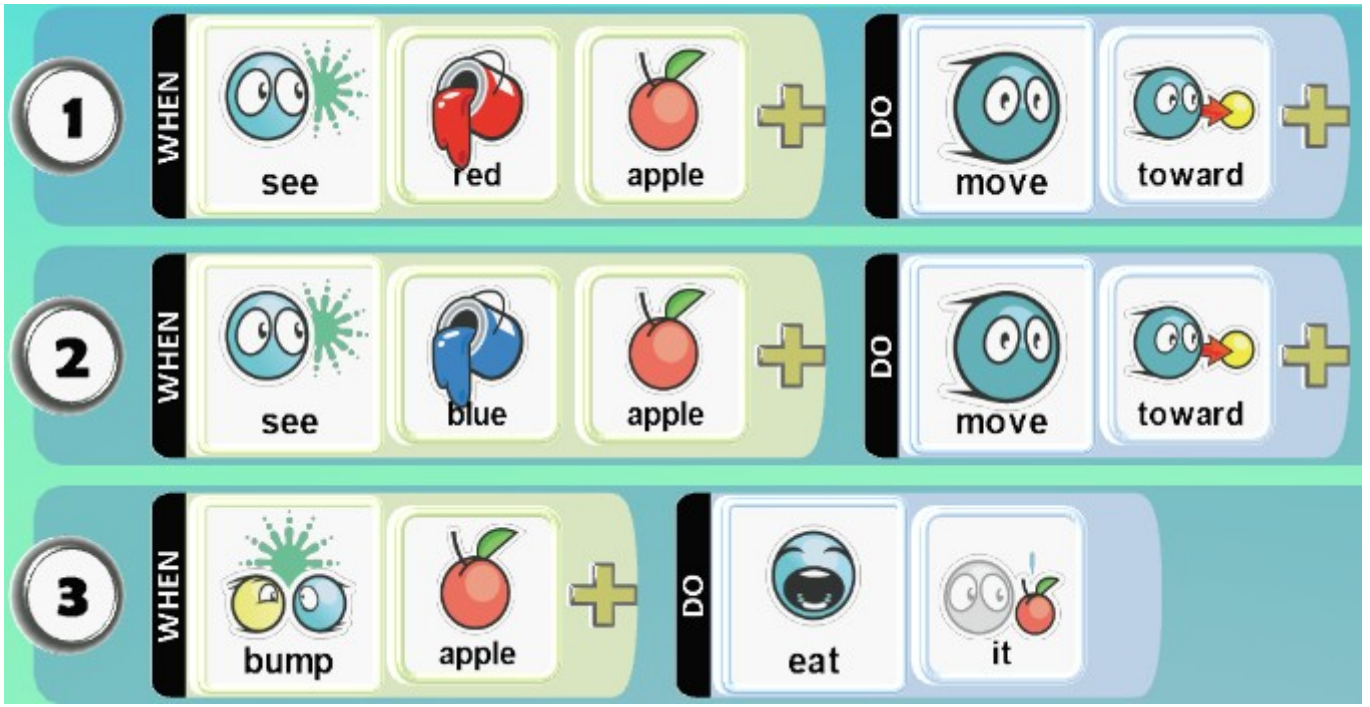
16/23 (70%) answered **1-4-3-2**.

2/23 answered **1-2-3-4** again: closest apple.

2/23 answered **1-2-4-3**.

Day 4 Q3

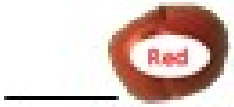




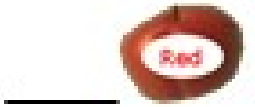
Why did 2/23 answer **1-2-4-3**, alternating red/blue?

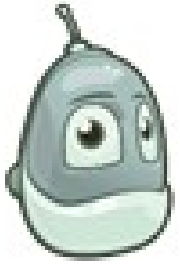
Hypothesis: they treated the rules as a sequential procedure.

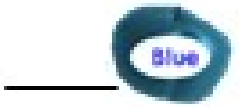
Day 4 Q4

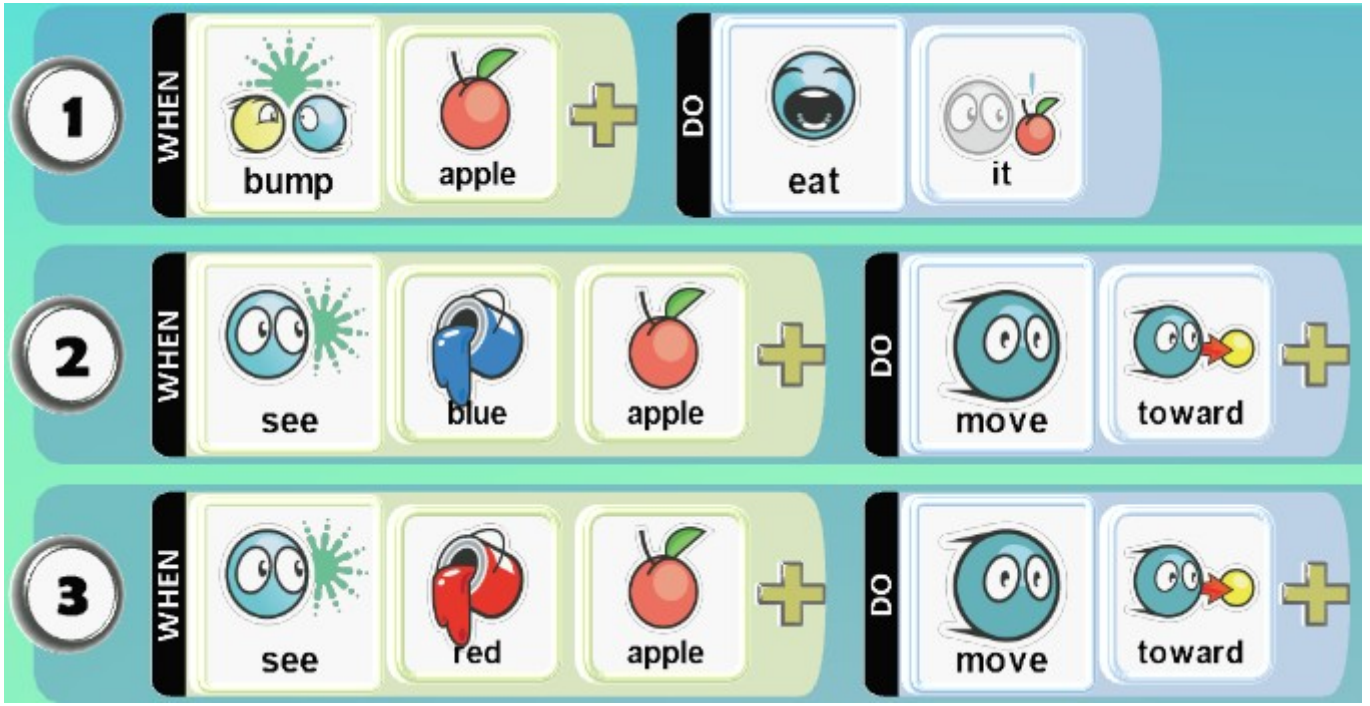

Red

Blue

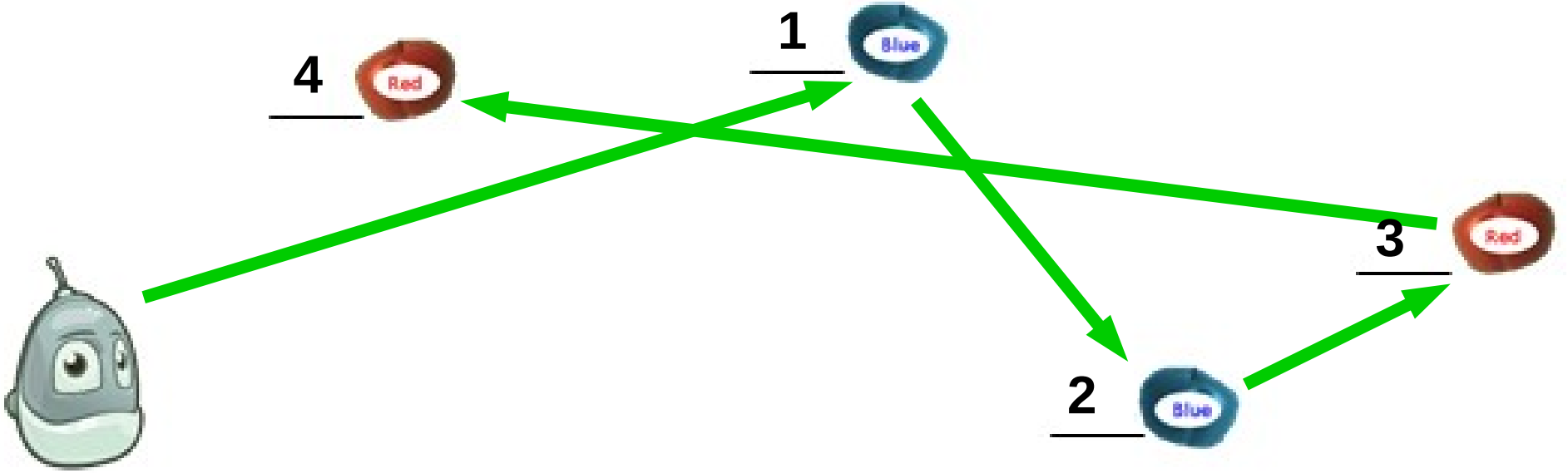

Red

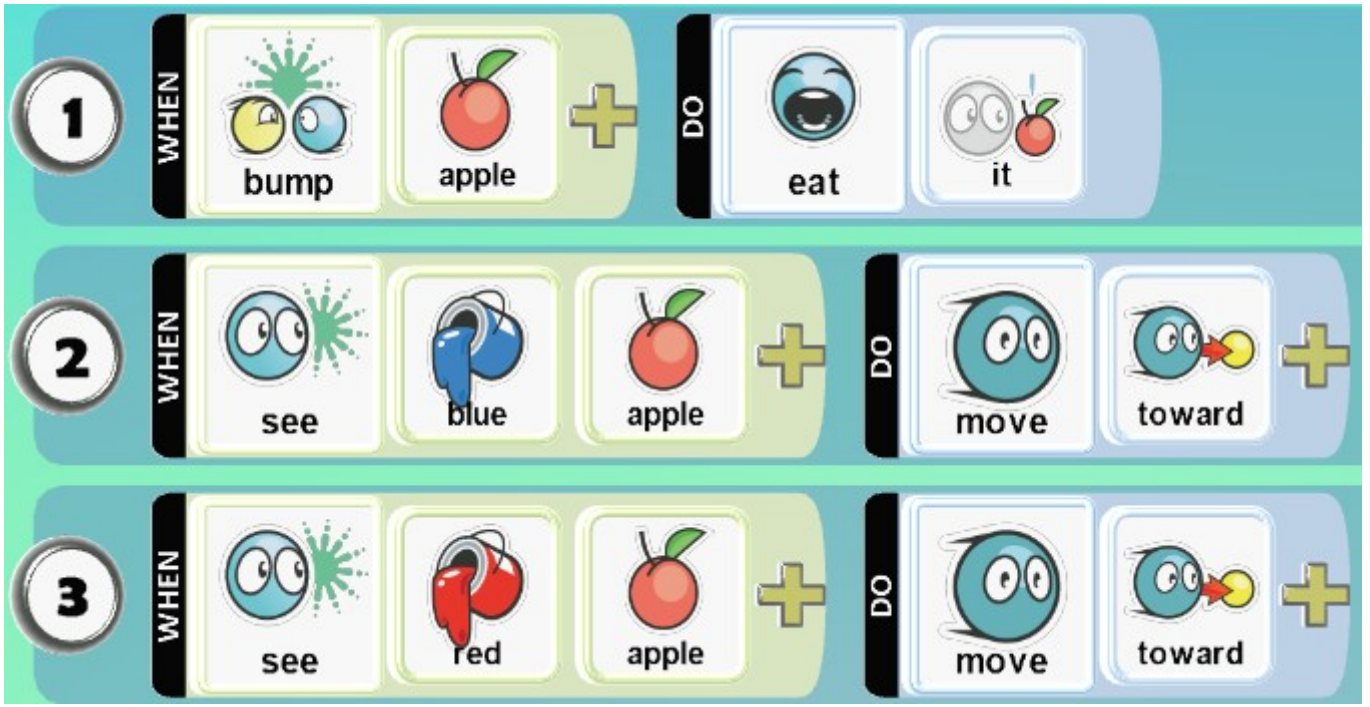



Blue


1
WHEN
bump
apple
DO
eat
it
2
WHEN
see
blue
apple
DO
move
toward
3
WHEN
see
red
apple
DO
move
toward

Day 4 Q4



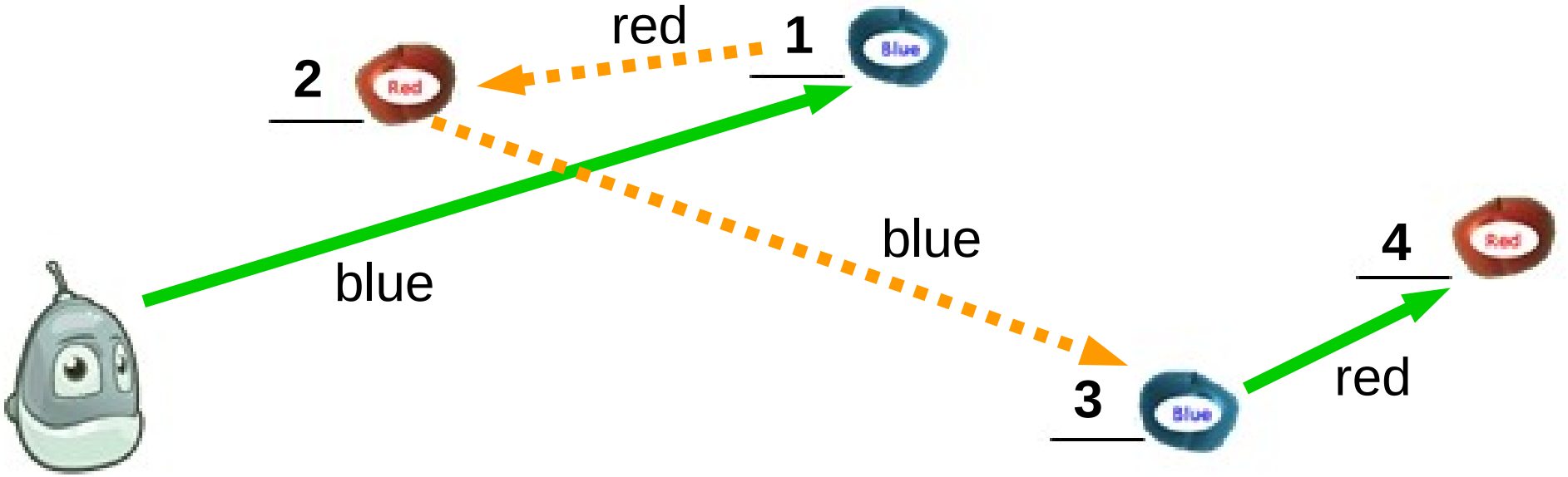


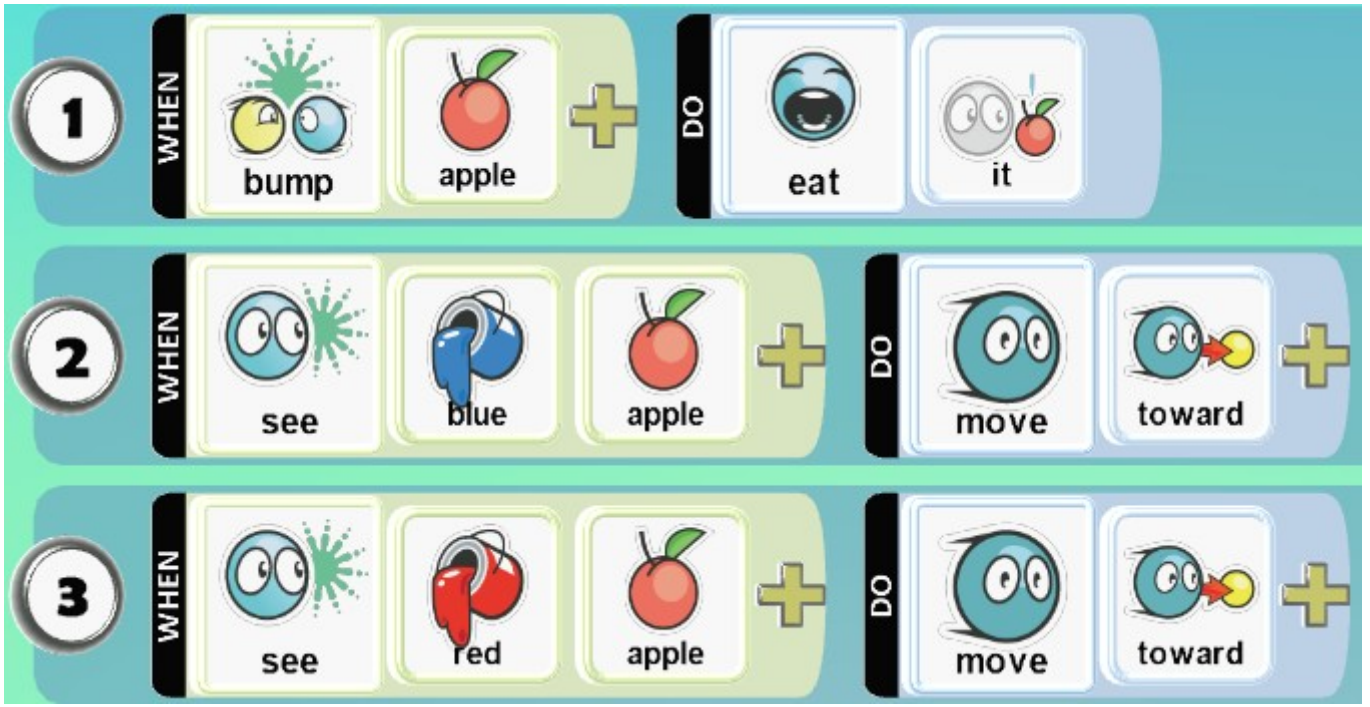
16/23 (70%) answered **4-1-2-3**.

2/23 answered **1-2-3-4** again.

2/23 answered **2-1-3-4**. Why?

Day 4 Q4





The 2/23 who answered **2-1-3-4** were alternating blue/red.

Same students who alternated red/blue on Q3.

More Abstract Reasoning About Rule Ordering

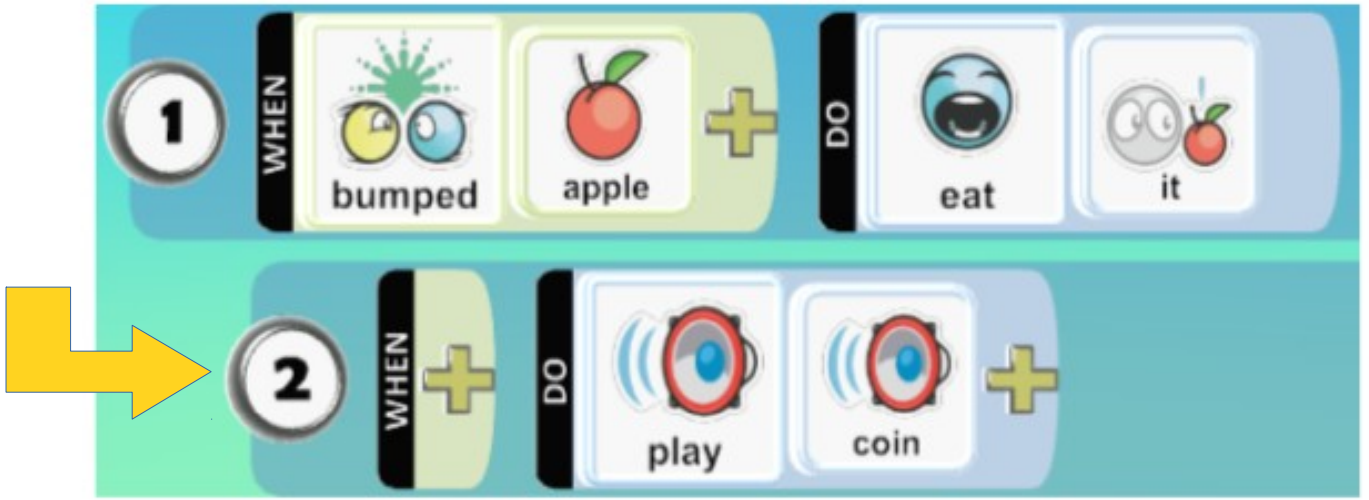
Sample questions (no images were provided):

- Compare *"Pursue and Consume"* with *"Default Value"*. Which idiom relies on rule ordering?
  - Only 8/23 (34%) answered correctly.
- Why does rule ordering matter for some idioms and not for others?
  - Only 5/23 (22%) gave an answer with some semblance of correctness.

Rule Dependency

Do Two Things

When you've bumped an apple, eat it *and also* play the coin sound.



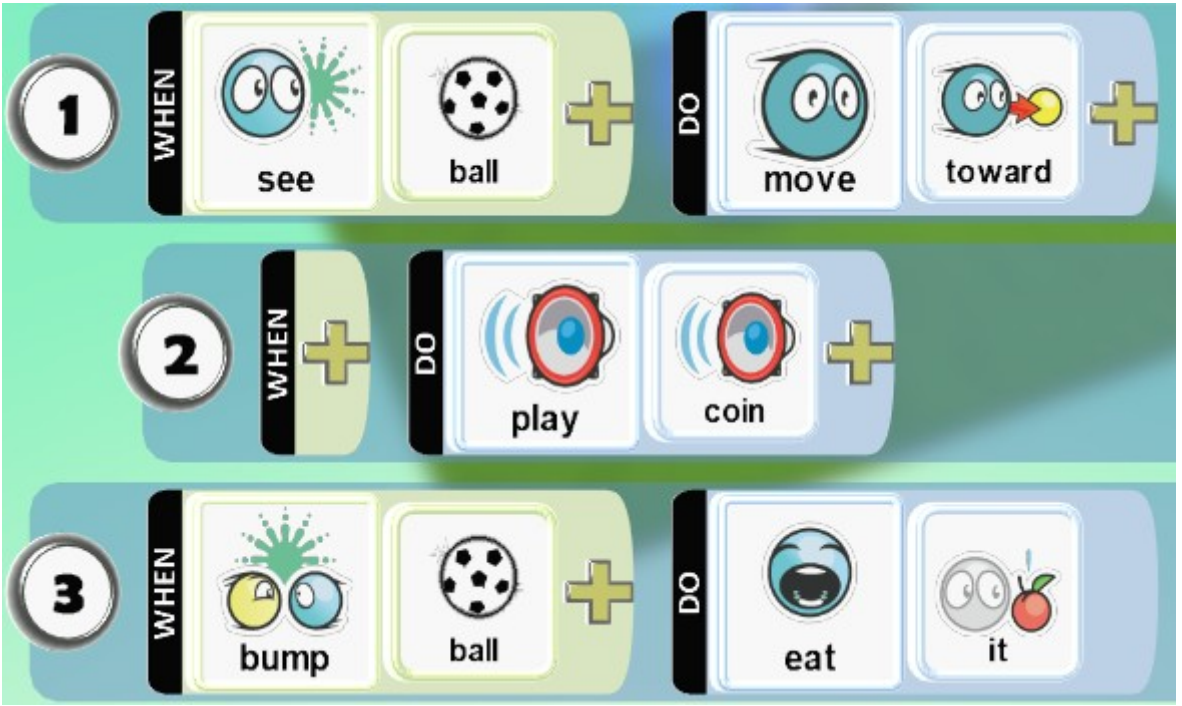
General Form:

WHEN *something* DO *action1*

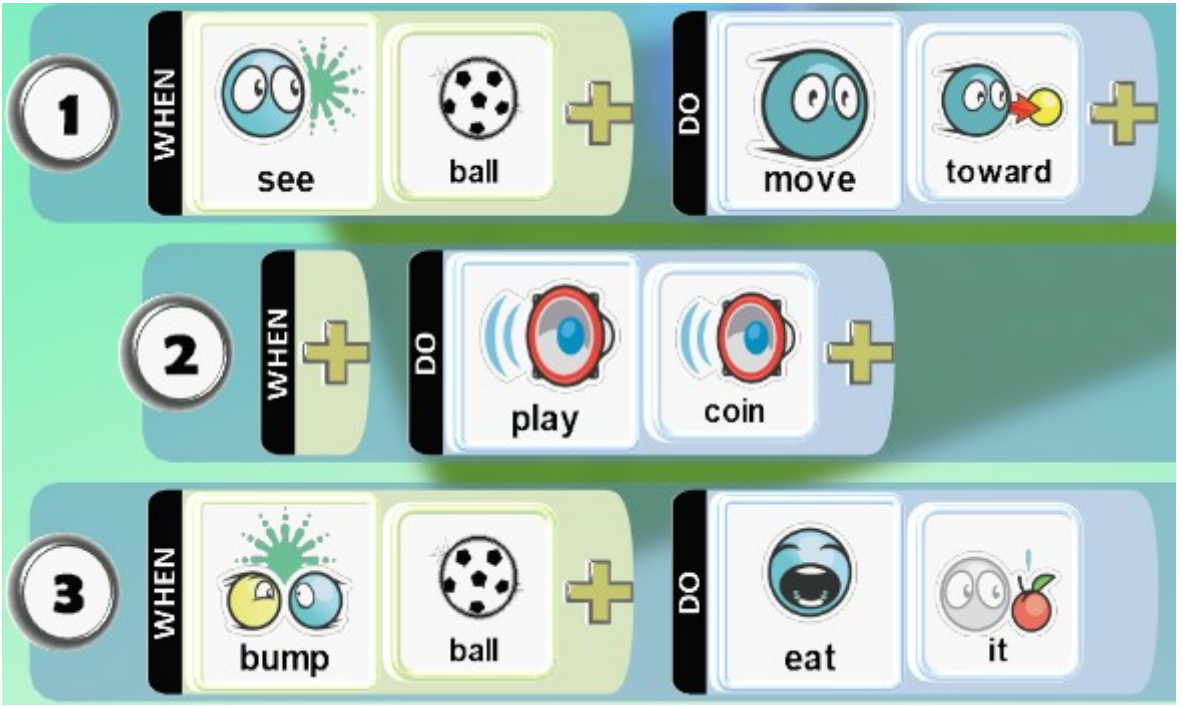
↳WHEN DO *action2*

Indenting the second rule makes it dependent on the WHEN part of the rule above.

When Will Kodu Play the Coin Sound?



When Will Kodu Play the Coin Sound?



18/23 (78%): "***When it sees the ball***" *or* "***When it moves forward***"

2/23: "***When it bumps the ball***"

3/23 gave **incoherent responses**.

Conclusions (1)

- Roughly 80% of students demonstrated an understanding of lawfulness in concrete situations.
  - They did less well on more abstract questions.

- Prior programming experience was **not** predictive of correct performance on the assessment questions on days 1-4. Possible explanations:
  - Kodu is very different from Scratch, Python, etc.
  - Students' earlier computing activities were not helping them appreciate lawfulness.

Conclusions (2)

Mastery of the fourth principle:

> *"When actions conflict, the lower numbered rule wins."*

Incorrect answers about **rule ordering effects** may reflect the misconception that a page of rules is a sequential procedure, as it would be in Scratch.

Conclusions (3)

Mastery of the third principle:

> *"An indented rule can run only if its parent's WHEN part is true."*

Incorrect answers about **rule dependency** may be a result of negative transfer from stereotypical examples, because the students were not exposed to atypical examples.

Conclusions (4)

- Our experiment identified two sources of misunderstanding that interfere with mastery.

- Kodu instructors should keep these sources of misunderstanding in mind when designing their curriculum:
  - Give more practice on rule ordering problems.
  - Have students practice with atypical examples before giving such examples in assessment tasks.