



# Exploring Novice Programmer Example Use

Michelle Ichinco and Caitlin Kelleher  
Washington University in St. Louis



# Programming as Assembly of Example Code

Programmers spend ~19% of their time programming on the web.

## Stack Overflow

21 Answers

▲ Use a sequential `for` loop:

1713

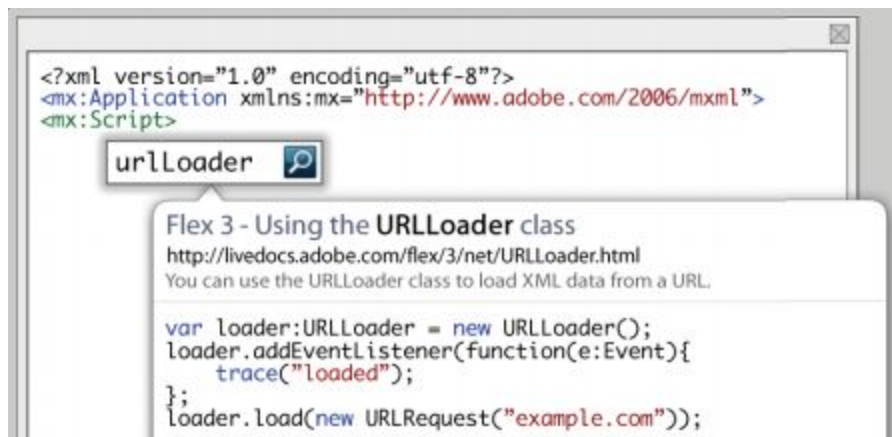
▼  

```
var myStringArray = ["Hello", "World"];
var arrayLength = myStringArray.length;
for (var i = 0; i < arrayLength; i++) {
    alert(myStringArray[i]);
    //Do something
}
```

## Java Doc Tutorials

```
printPersonsWithPredicate(
    roster,
    p -> p.getGender() == Person.Sex.MALE
        && p.getAge() >= 18
        && p.getAge() <= 25
);
```

# Existing Example Support



# Novice Programmers Struggle Using Examples

Non-expert programmers have trouble reusing code found online without support

- Messing up working code
- Code not accomplishing goal

# Example Use Studies Focus on Adult Programmers

Increasing number of programming applications and resources for children



1



Looking Glass


2

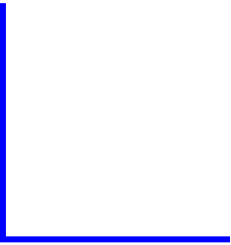


3

1. “Anybody Can Learn | Code.org” [Online]. Available: <http://code.org>
2. “Looking Glass Community.” [Online]. Available: <https://lookingglass.wustl.edu/>.
3. “Scratch | Home | imagine, program, share.” [Online]. Available: <http://scratch.mit.edu/>

# Research Questions



1. What **hurdles** do novice programmers encounter using examples?
  2. What **strategies** do novice programmers use while attempting to use examples?
- 

- Exploratory study
- Labelled transcript data to understand behavior
- Hurdles & Strategies

# User Study





# What does novice example use look like?

*“I don’t like to copy because I like to learn how to do it myself.”*



# Novice Example Use

The screenshot shows the Scratch IDE interface. At the top, there's a toolbar with buttons for 'Create or Open World', 'Save', 'Undo', 'Redo', 'Play', and 'Find A Remix'. On the right, there are share options: 'World', 'Remix', and 'Template'.

The main workspace is titled 'Scene' and 'My Story'. Below this, a 'custom action My Story' is defined. The 'Do in order' section contains five steps:

- Pig say "I'm here!! I'm here!!" add detail
- Pig getRightShoulder turn BACKWARD, 0.25 rotations add detail
- Pig getRightShoulder turn FORWARD, 0.25 rotations add detail
- Pig getLeftShoulder turn BACKWARD, 0.25 rotations add detail
- Pig getLeftShoulder turn FORWARD, 0.25 rotations add detail

On the left, there's a 'Pig's Actions' panel with categories: 'walk', 'say, think', 'position', 'Pig's Questions', and 'Action Ordering Boxes'. Each category contains several action blocks for the pig character.

On the right, a 'Task and Example' panel is displayed. It contains the following text:

**Task** Make the pig wave both of his hands up and then both down to signal the helicopter.

**instructions:**

**Use this example to help you solve the task:**

**Do together**

- Dolphin move UP, 2.5 meters add detail
- Dolphin turn BACKWARD, 0.125 rotations add detail
- Dolphin move FORWARD, 2.5 meters add detail
- Dolphin turn FORWARD, 0.125 rotations add detail

At the bottom of the 'Task and Example' panel is a 'Play Example' button.

# Do together task

Task



Script for the pig-like creature:

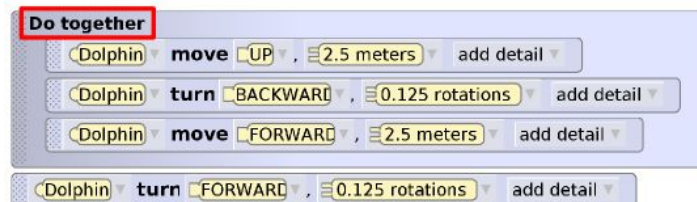
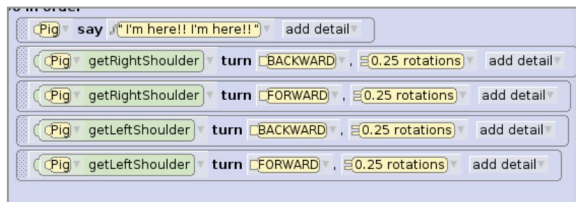
- Pig say "I'm here!! I'm here!!" add detail
- Pig getRightShoulder turn BACKWARD, 0.25 rotations add detail
- Pig getRightShoulder turn FORWARD, 0.25 rotations add detail
- Pig getLeftShoulder turn BACKWARD, 0.25 rotations add detail
- Pig getLeftShoulder turn FORWARD, 0.25 rotations add detail

Example

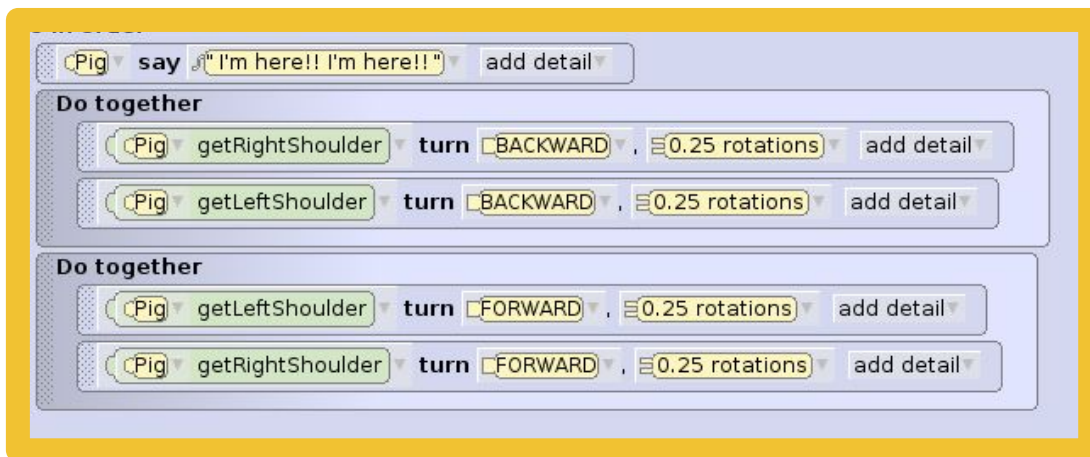
**Do together**


- Dolphin move UP, 2.5 meters add detail
- Dolphin turn BACKWARD, 0.125 rotations add detail
- Dolphin move FORWARD, 2.5 meters add detail
- Dolphin turn FORWARD, 0.125 rotations add detail

# Do together task

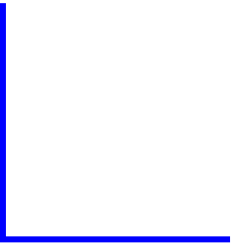


Completed  
Task

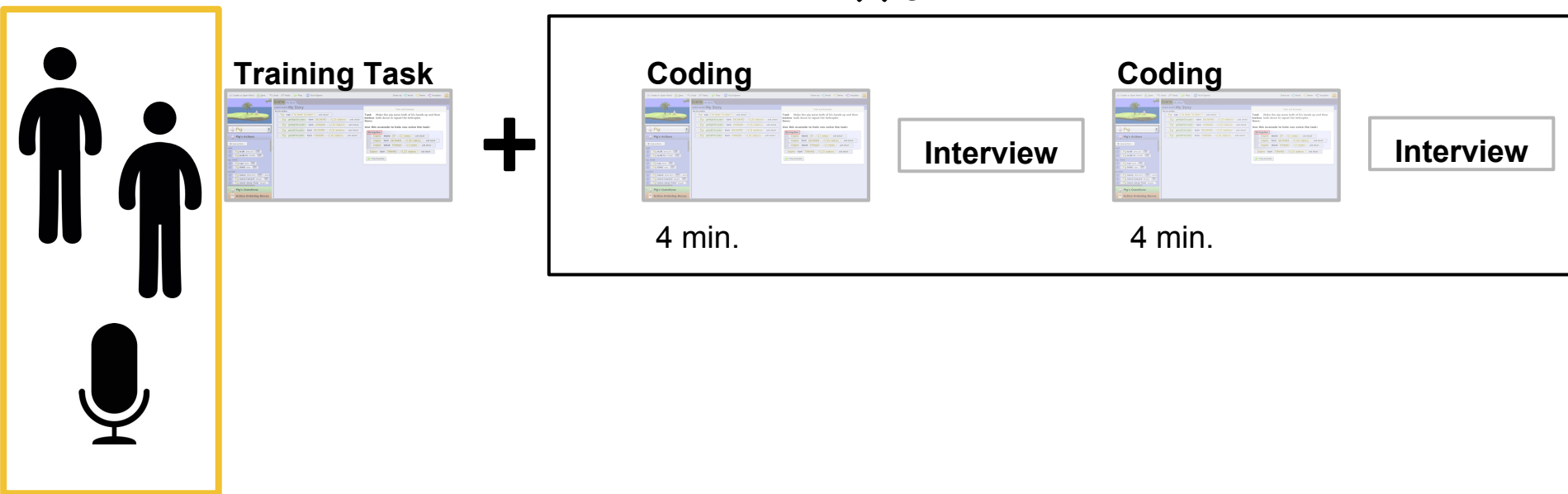




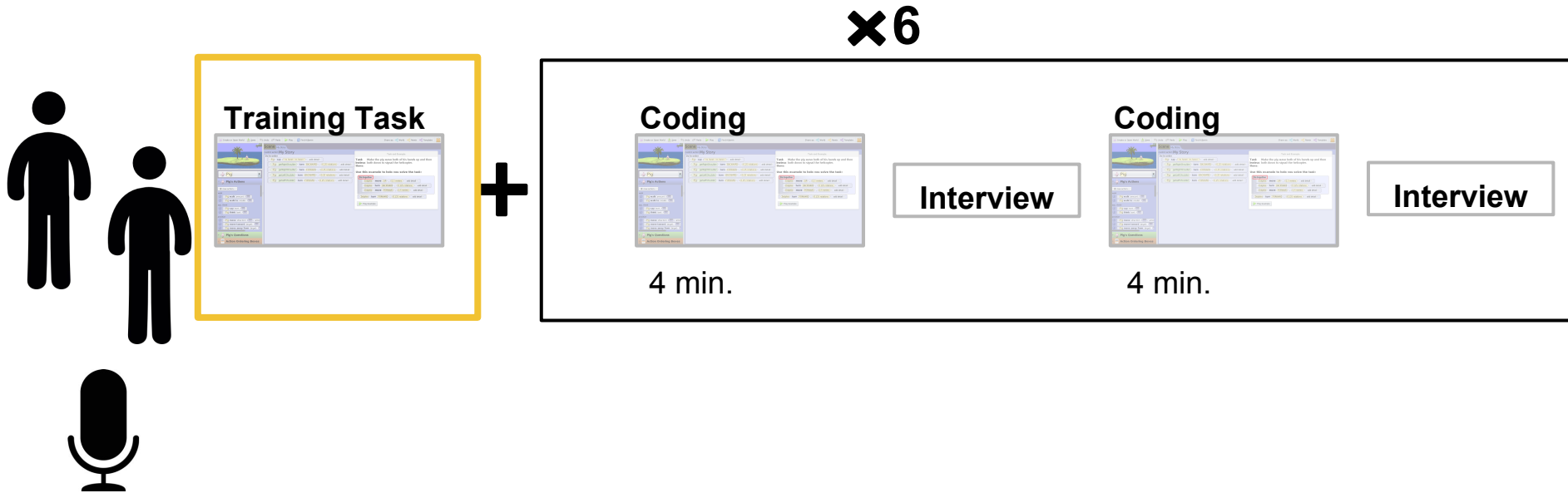
What are novices thinking while completing tasks with examples?



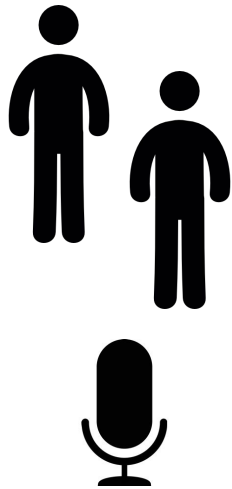
# Study Format



# Study Format



# Study Format



## Training Task



+

×6

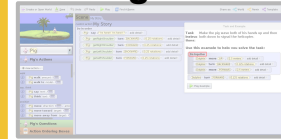
## Coding



4 min.

Interview

## Coding



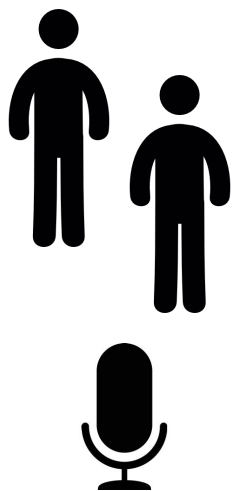
4 min.

Interview

- 6 concepts: simple parallel execution, for loop, API method, function, while loop condition, for each iterator
- Based on formative and pilot studies



# Study Format



**Training Task**



+

**×6**

**Coding**



4 min.

**Interview**

**Coding**



4 min.

**Interview**

# Study Participants


## **9 pairs of participants**

Participant screening criteria:

aged 10-15

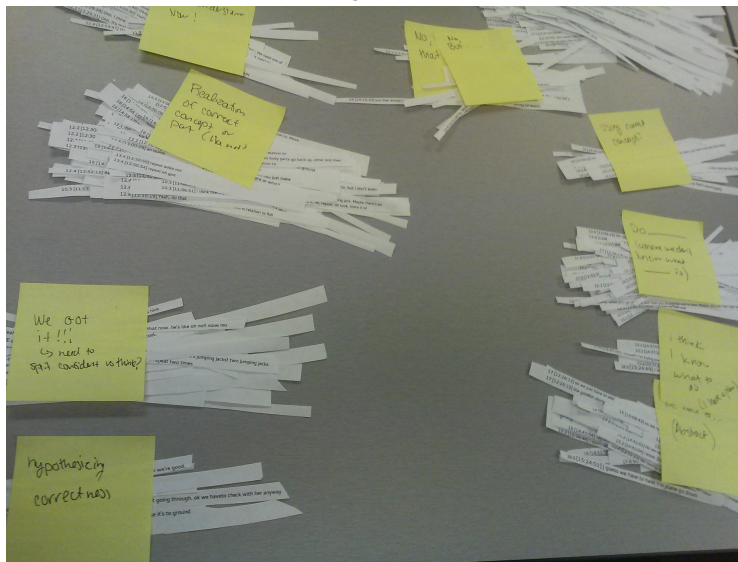
$\leq$  3 hours of programming experience

# Research Questions



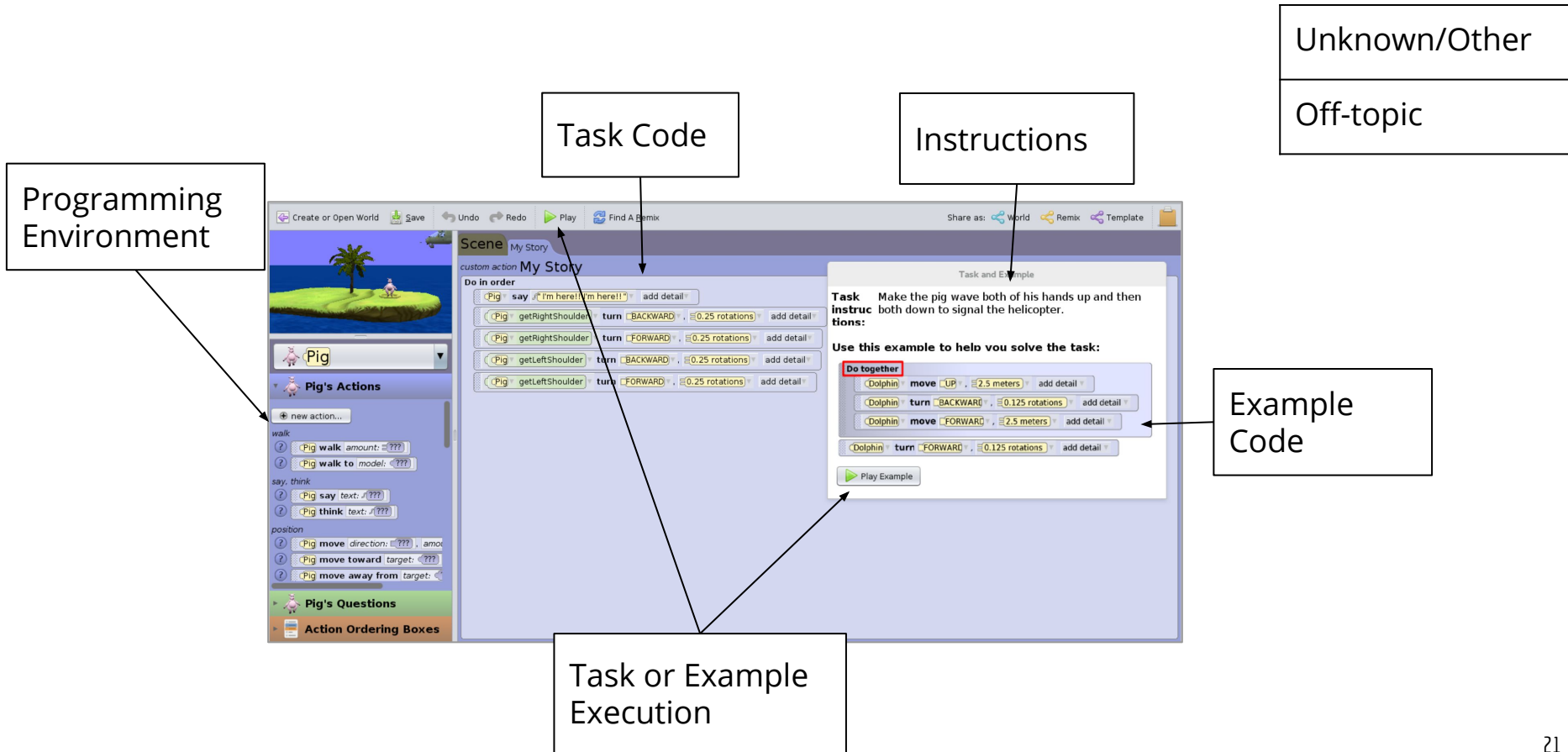
1. What **hurdles** do novice programmers encounter using examples?
2. What **strategies** do novice programmers use while attempting to use examples?

# Qualitative Analysis: Transcription Labeling



- Inter-rater agreement: 20% of transcripts, **83% agreement**
- Two groups:
  - Focus Area: which part of the task/environment
  - Process: what they're doing

# Focus area labels



# Focus area labels

## Programming Environment:

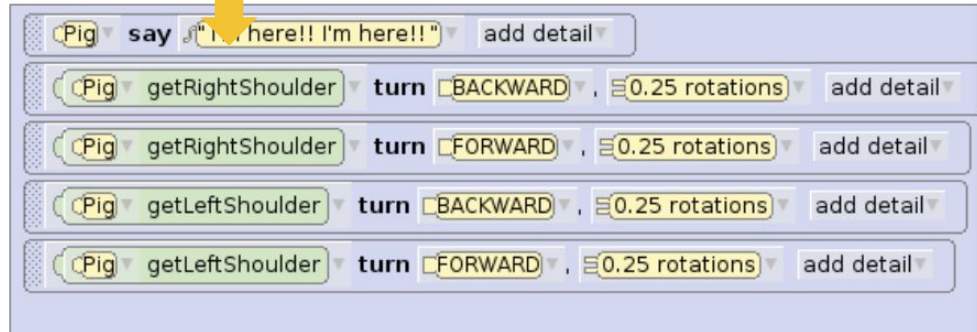
*"Pig actions Place da da da da . Turn. Appear disappear, resize. Hmm"*



# Focus area labels

## Task Code:

*"There's get right shoulder. hmmm. Get right shoulder hmmm"*



# Process labels

- Description
- Idea
- Evaluation
- Unknown/Other



# Process labels

- **Description**
- Description-realization
- Description-don't understand



***“Play the example again cause. It goes up and then it stops and turns a bit”***

# Process labels

**Example execution:  
description**



***“Play the example again cause. It goes up and then it stops and turns a bit”***

# Process labels

- **Idea**
- Idea-realization
- Idea-don't know how



***“Maybe I could try using two of them [do togethers] instead.”***

# Process labels

Task code: idea



***“Maybe I could try using two of them [do togethers] instead.”***

# Process labels

- **Evaluation-working**
- Evaluation-possibly working
- Evaluation-not working

***“I'm just gonna see if this works.YESS! We did it!!! awesome!!!”***

# Realization labels

*“OHHHHH! So I don't know where we'd get this whole repeat 4 times thing, but I think I kind of understand it.”*

# Realization labels

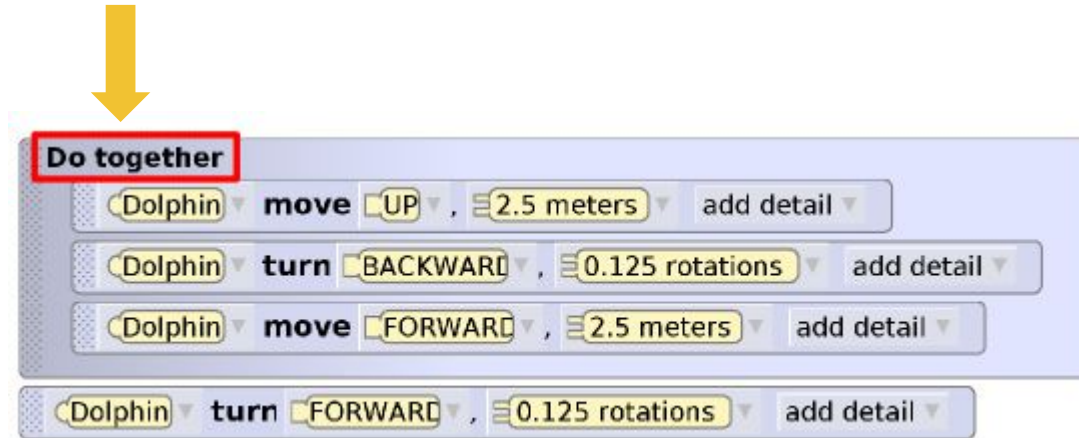
*“OHHHHH! So I don't know where we'd get this whole repeat 4 times thing, but I think I kind of understand it.”*

*“Ohh, so jump. Jumping jack. Then you do repeat two times”*

*“Wait a minute. wait can I see that again? STOP! OHH cause it's like. here, give me the mouse.”*

# Realization Point: the “first” realization

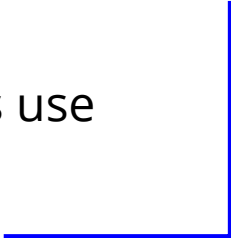
**Definition: the point in time when one participant first talks about the concept in the example needed to complete the task**







# Results

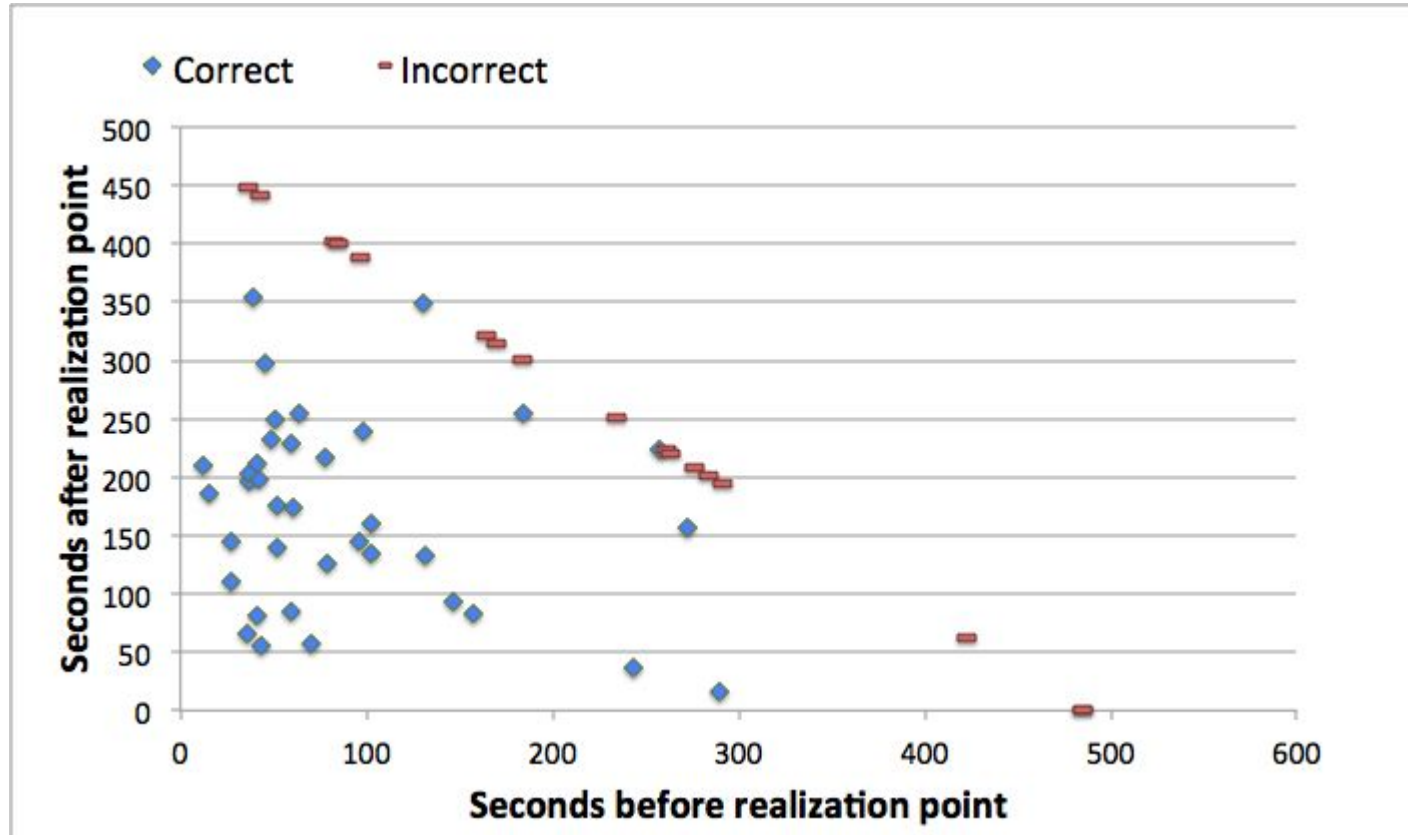
1. What **hurdles** do novice programmers encounter using examples?
  2. What **strategies** do novice programmers use while attempting to use examples?
- 



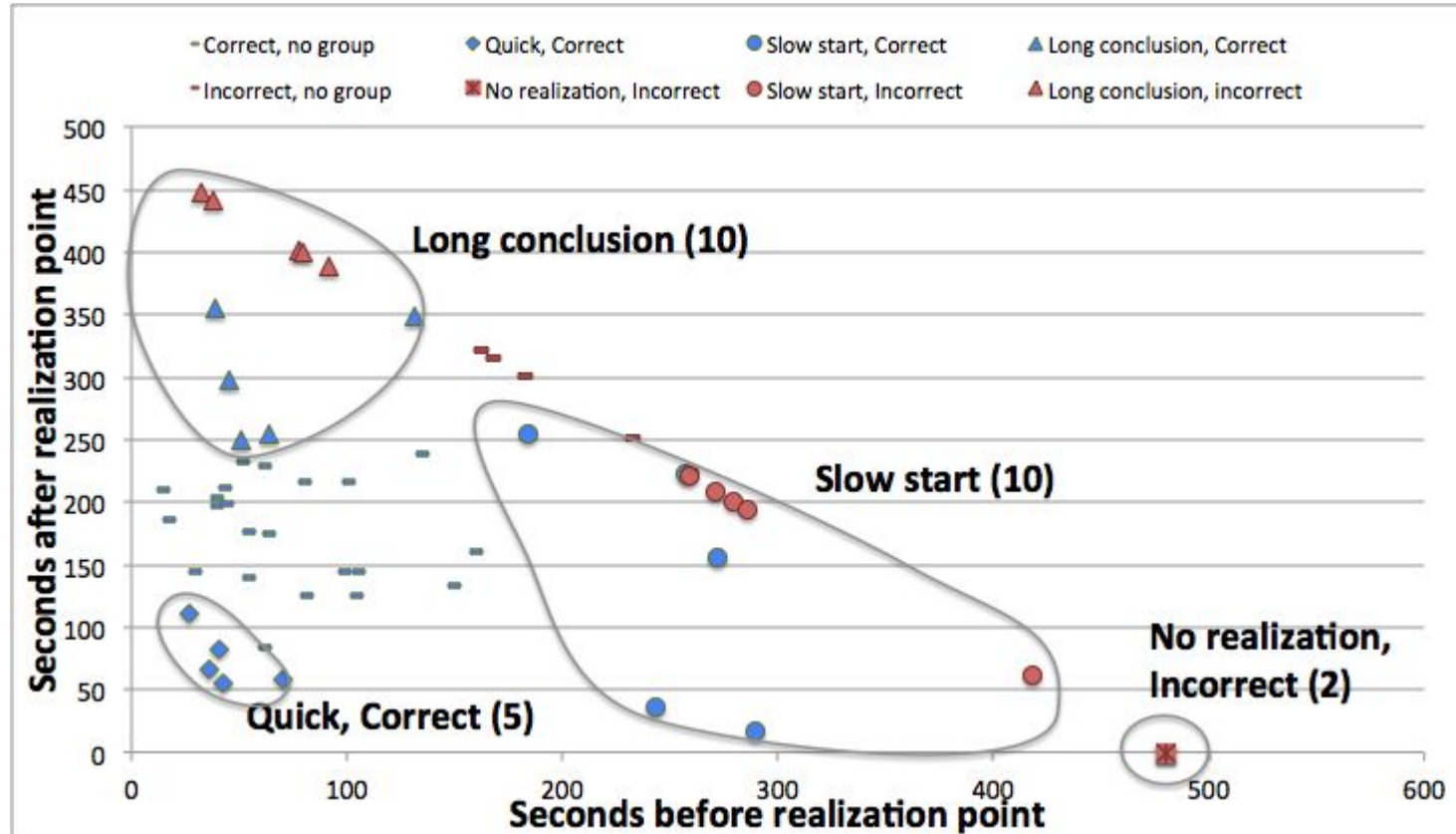
How can we identify strategies and hurdles?



# Task Times Before and After Realization Point



# Task Behavior Groups



# Hurdles & Strategies

## Hurdles:

- Example Comprehension
- Programming Environment
- Code Comprehension
- Context Distraction
- Code Misconception

## Strategies:

- Idea Generation
- Code-Example Comparison
- Example Emphasis

# Hurdles

# Example Comprehension Hurdle

“Play example. I don't get how that's supposed to help us. Yeah, I have no idea.”

# Example Comprehension Hurdle

“Play example. I don't get how that's supposed to help us. Yeah, I have no idea.”

The screenshot displays a programming environment with a top toolbar containing icons for 'Create or Open World', 'Save', 'Undo', 'Redo', 'Play', and 'Find A Remix'. On the right, there are 'Share as' options: 'World', 'Remix', and 'Template'.

The main workspace is titled 'Scene My Story' and shows a custom action 'My Story' with a 'Do in order' sequence of five blocks for a pig:

- Pig say "I'm here!! I'm here!!" add detail
- Pig getRightShoulder turn BACKWARD, 0.25 rotations add detail
- Pig getRightShoulder turn FORWARD, 0.25 rotations add detail
- Pig getLeftShoulder turn BACKWARD, 0.25 rotations add detail
- Pig getLeftShoulder turn FORWARD, 0.25 rotations add detail

On the left sidebar, there is a 'Pig' character selection, a 'Pig's Actions' list with categories like 'walk', 'say, think', and 'position', and 'Pig's Questions' and 'Action Ordering Boxes' at the bottom.

On the right, a 'Task and Example' panel is shown. It contains a 'Task' description: 'Make the pig wave both of his hands up and then both down to signal the helicopter.' Below this is an 'Instructions' section. A 'Do together' box highlights a sequence of four blocks for a dolphin:

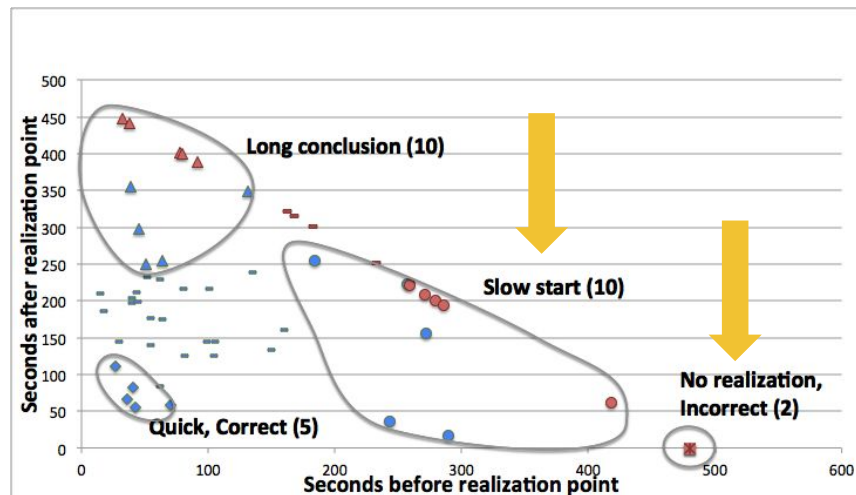
- Dolphin move UP, 2.5 meters add detail
- Dolphin turn BACKWARD, 0.125 rotations add detail
- Dolphin move FORWARD, 2.5 meters add detail
- Dolphin turn FORWARD, 0.125 rotations add detail

A 'Play Example' button is located at the bottom of the 'Do together' box.



# Example Comprehension Hurdle

Label	No realization	Slow start, incorrect
Execution Example Code	.8 ✖ average (BR)	1.6 ✖ average (BR)
Example Code	0 ✖ average (BR)	1.6 ✖ average (BR)

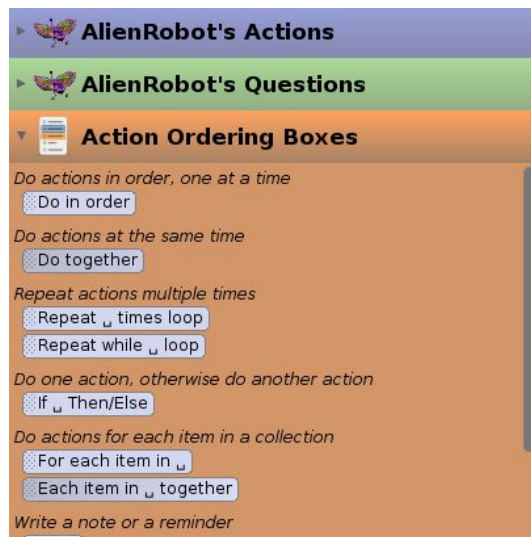
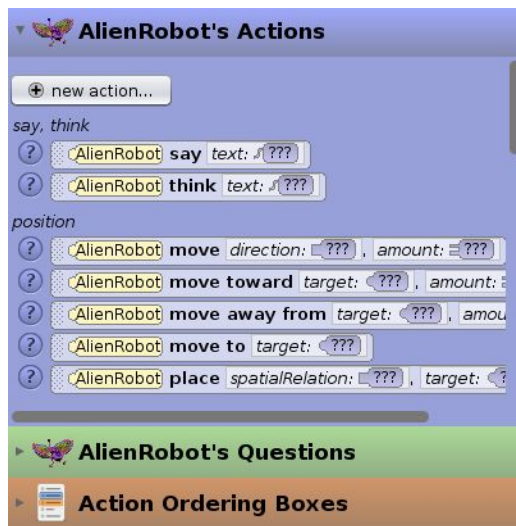


# Programming Environment Hurdle

“Then you do repeat two times. How? But it says that you can repeat. **Where is the times thing? I don’t see that.** ”

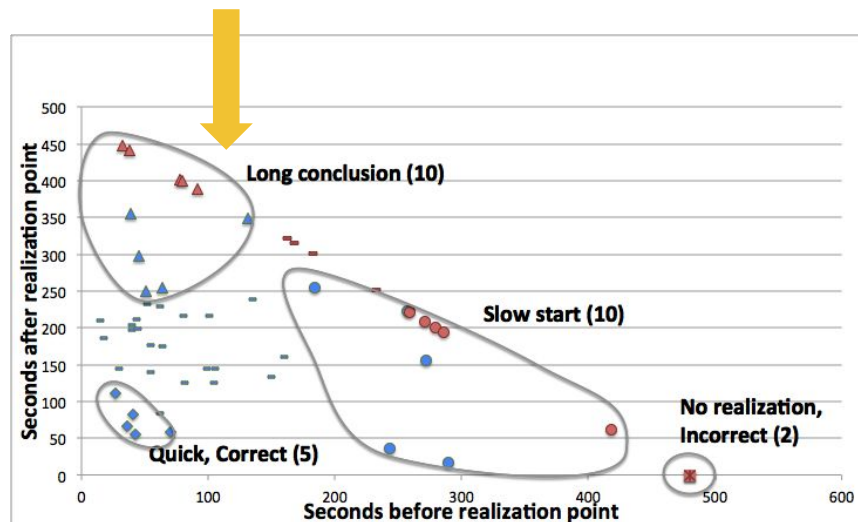
# Programming Environment Hurdle

“Then you do repeat two times. How? But it says that you can repeat. **Where is the times thing? I don't see that.** ”



# Programming Environment Hurdle

Label	Long Conclusion, incorrect
Programming Environment: description- don't understand Programming Environment: idea- don't know how	3.6 ✖ average (AR)



# Code Comprehension Hurdle

“Why is he not on the ground?”

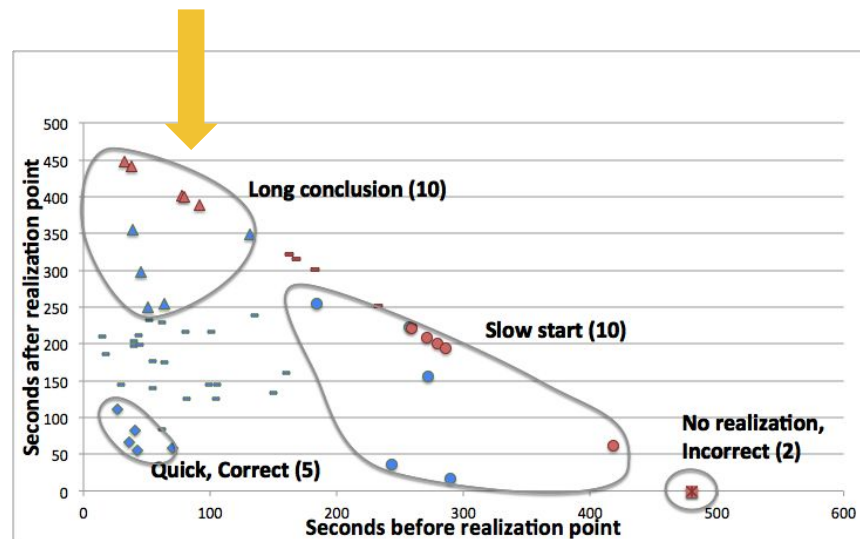
# Code Comprehension Hurdle

“Why is he not on the ground?”



# Code Comprehension Hurdle

Label	Long Conclusion, incorrect
Task execution: description- don't understand	3.7 ✖ average (AR)



# Strategies

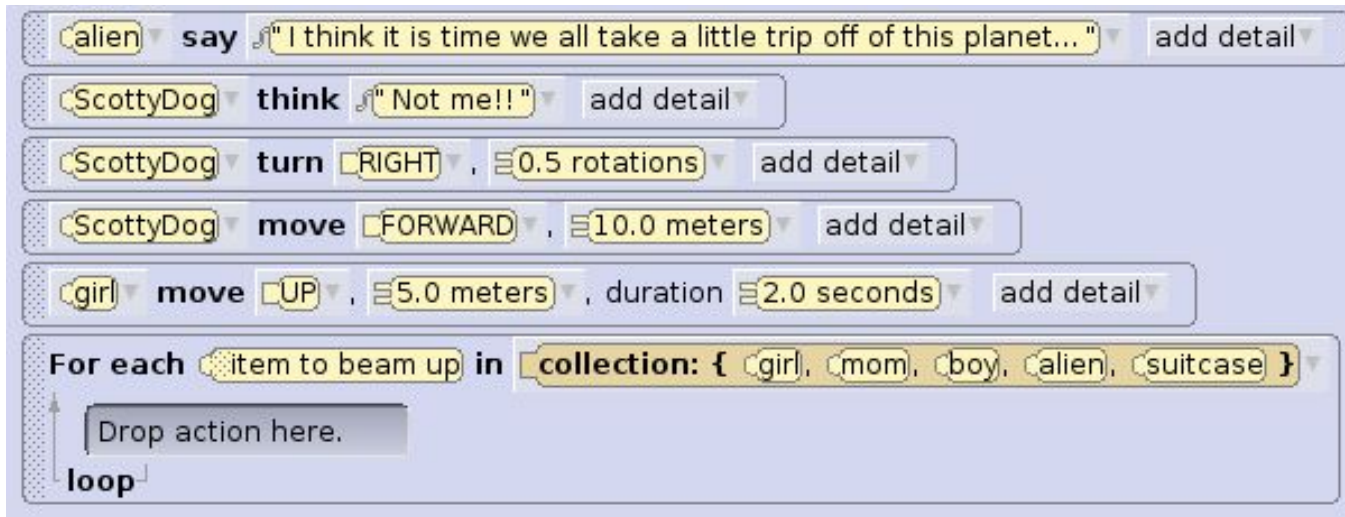


# Idea Generation Strategy

“Do we have to put that up there or what? Do we move them in there or something? For it to work? Do we move this?”

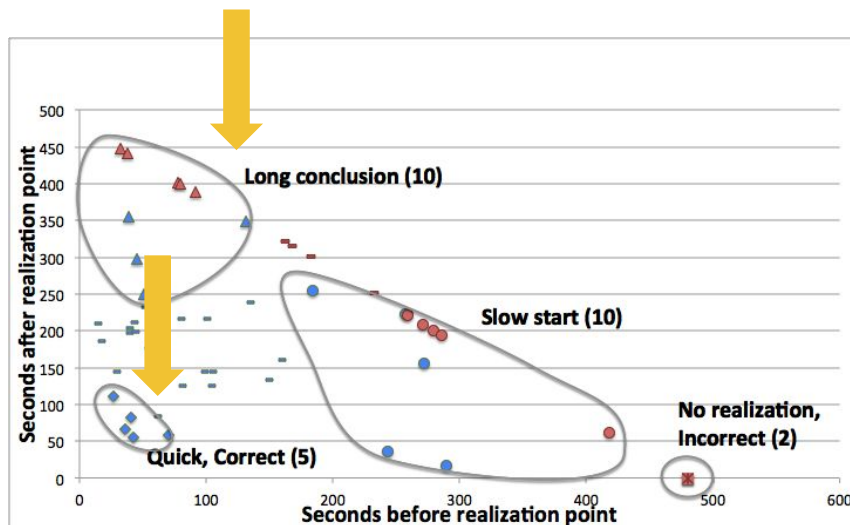
# Idea Generation Strategy

“Do we have to put that up there or what? Do we move them in there or something? For it to work? Do we move this?”



# Idea Generation Strategy

Label	Long Conclusion,correct	Quick, correct
Task Code: idea	1.25 ✖ average (AR)	.4 ✖ average (AR)
Task Code Execution	1.1 ✖ average (AR)	.8 ✖ average (AR)
Task Code: Description- don't understand	.5 ✖ of average (AR)	.5 ✖ of average (AR)

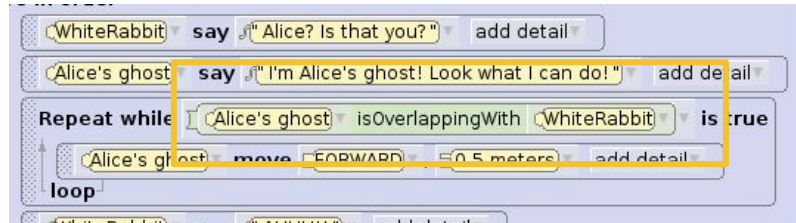


# Code-Example Comparison Strategy

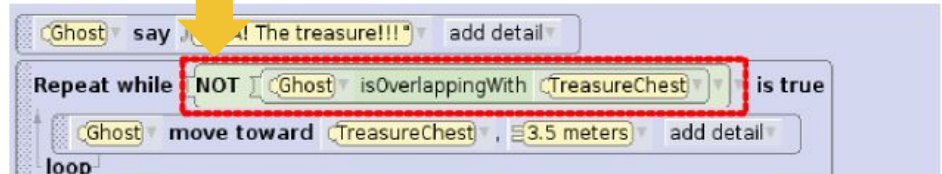
“Not is true. But here it’s just is true ... That looks like the example [...] but it’s got this not thing.”

# Code-Example Comparison Strategy

“Not is true. But here it's just is true ... That looks like the example [...] but it's got this not thing.”

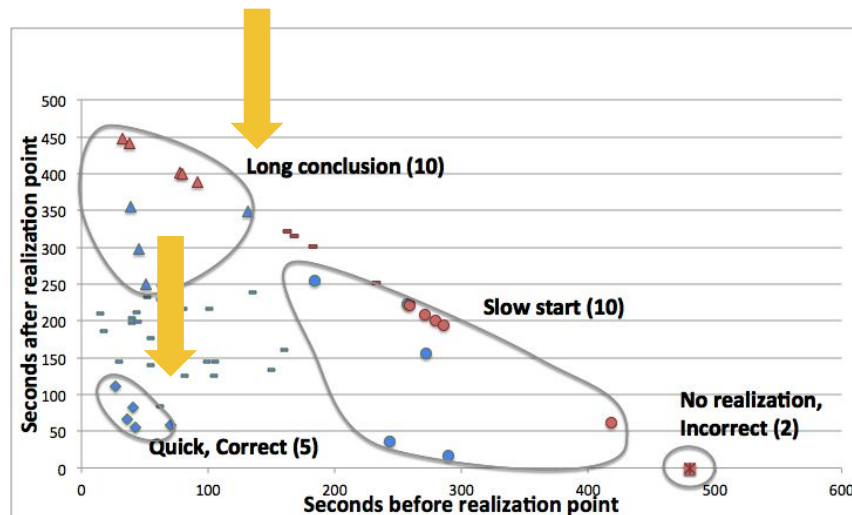


Use this example to help you solve the task:



# Code-Example Comparison Strategy

Label	Long Conclusion, correct	Quick, correct
Example Code	1.4 ✖ average (AR)	.75 ✖ average (AR)





# What can we do with these hurdles and strategies?

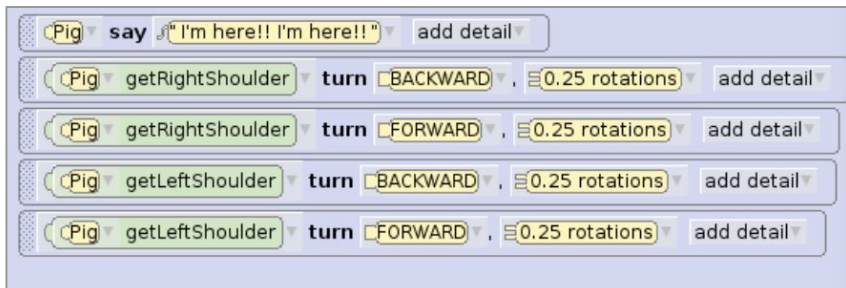
Educational Systems • Visual Programming Systems • General Example Use



# For Educational Systems

Certain behavior can indicate good times for novices to return to examples.

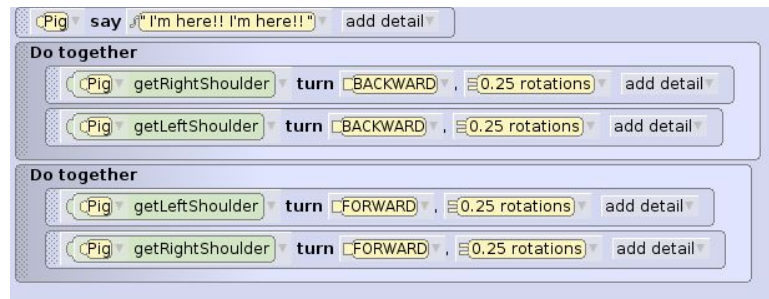
## Task Code



Scratch code for the task. The code consists of five blocks:

- say** "I'm here!! I'm here!!" for 2 seconds, then **add detail**.
- getRightShoulder** **turn** **BACKWARD** by 0.25 rotations, then **add detail**.
- getRightShoulder** **turn** **FORWARD** by 0.25 rotations, then **add detail**.
- getLeftShoulder** **turn** **BACKWARD** by 0.25 rotations, then **add detail**.
- getLeftShoulder** **turn** **FORWARD** by 0.25 rotations, then **add detail**.

## Correct Solution



Scratch code for the correct solution. The code consists of five blocks, organized into two groups:

- say** "I'm here!! I'm here!!" for 2 seconds, then **add detail**.
- Do together** group containing:
  - getRightShoulder** **turn** **BACKWARD** by 0.25 rotations, then **add detail**.
  - getLeftShoulder** **turn** **BACKWARD** by 0.25 rotations, then **add detail**.
- Do together** group containing:
  - getLeftShoulder** **turn** **FORWARD** by 0.25 rotations, then **add detail**.
  - getRightShoulder** **turn** **FORWARD** by 0.25 rotations, then **add detail**.



# For Educational Systems

Certain behavior can indicate good times for novices to return to examples.

Examples of such behaviors include:

- many interface interactions that do not lead to solution (BR)
- many incorrect attempts or repeated attempts with minor modifications (AR)

# For visual programming environments

It is important to enable novices to easily access code elements in examples.

The image displays a visual programming interface. On the left, a 'Do together' block is highlighted with a red border. It contains four parallel action blocks, each starting with a 'Dolphin' object. The actions are: 'move UP, 2.5 meters', 'turn BACKWARD, 0.125 rotations', 'move FORWARD, 2.5 meters', and 'turn FORWARD, 0.125 rotations'. Each action block has an 'add detail' button. A yellow arrow points from the 'Do together' block to a library on the right. The library is titled 'AlienRobot's Actions' and 'AlienRobot's Questions'. Below these, a section titled 'Action Ordering Boxes' lists various control structures: 'Do in order', 'Do together', 'Repeat times loop', 'Repeat while loop', 'If Then/Else', 'For each item in', and 'Each item in together'. The 'Do together' box is highlighted in the library.

**Do together**

Dolphin move UP, 2.5 meters add detail

Dolphin turn BACKWARD, 0.125 rotations add detail

Dolphin move FORWARD, 2.5 meters add detail

Dolphin turn FORWARD, 0.125 rotations add detail

**AlienRobot's Actions**

**AlienRobot's Questions**

**Action Ordering Boxes**

Do actions in order, one at a time

Do in order

Do actions at the same time

Do together

Repeat actions multiple times

Repeat times loop

Repeat while loop

Do one action, otherwise do another action

If Then/Else

Do actions for each item in a collection

For each item in

Each item in together

Write a note or a reminder

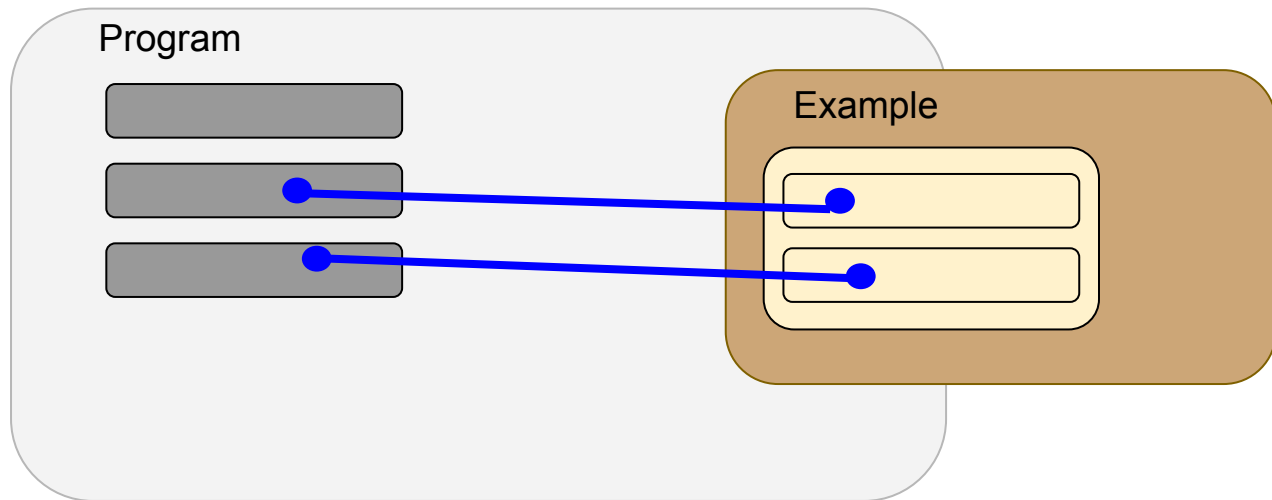
# For visual programming environments

It is important to enable novices to easily access code elements in examples.

- visual cues
- functionality to improve and expedite access

# General Example Use

How can we help novices better **understand and utilize** the **relationships between examples and program code**?



# Thank You

michelle.ichinco@wustl.edu



# Context Distraction Hurdle

“Wait, can you, wait click ‘as seen by’ just out of curiosity”

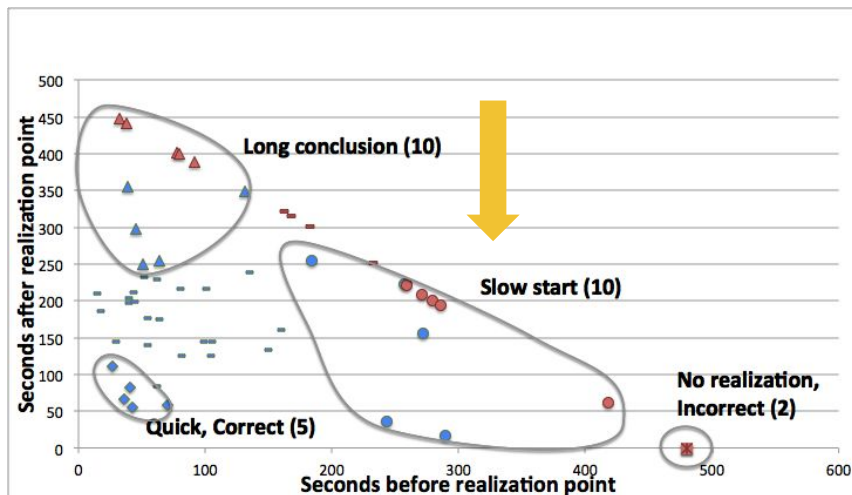
# Context Distraction Hurdle

“Wait, can you, wait click ‘as seen by’ just out of curiosity”



# Context Distraction Hurdle

Label	Slow start, correct
Programming Environment: idea	3.75 ✖ average (BR)
Instructions: description don't understand	4.3 ✖ average (BR)





# Summary

- Novice programmers may need time to get familiar with task, example, environment before example is useful to them
- Example is often used as a starting point
  - use idea generation to try out new plans
  - returning to the example can help
  - programming environment and lack of further assistance can slow down or impede process

# Focus area labels

Instructions

The screenshot shows the Scratch 'My Story' workspace. On the left, the 'Pig's Actions' sidebar lists various actions like 'walk', 'say', 'think', 'position', and 'Pig's Questions'. The main workspace displays a custom action for the 'Pig' character, titled 'My Story'. The 'Do in order' section contains a sequence of actions: 'Pig say "I'm here!! I'm here!!"', 'Pig getRightShoulder turn BACKWARD 0.25 rotations', 'Pig getRightShoulder turn FORWARD 0.25 rotations', 'Pig getLeftShoulder turn BACKWARD 0.25 rotations', and 'Pig getLeftShoulder turn FORWARD 0.25 rotations'. On the right, the 'Task and Example' panel shows a task: 'Make the pig wave both of his hands up and then both down to signal the helicopter.' Below the task, it says 'Use this example to help you solve the task:' and provides a 'Do together' example with actions for a 'Dolphin' character: 'move UP 2.5 meters', 'turn BACKWARD 0.125 rotations', 'move FORWARD 2.5 meters', and 'turn FORWARD 0.125 rotations'. A box labeled 'Instructions' points to the task text.

Task: Make the pig wave both of his hands up and then both down to signal the helicopter.

Instructions:

Use this example to help you solve the task:

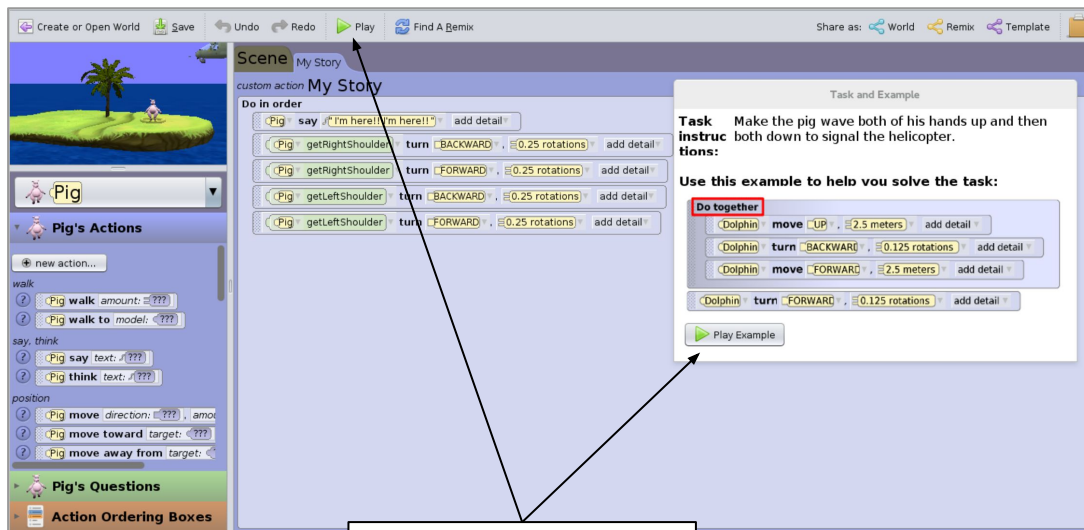
**Do together**

- Dolphin move UP 2.5 meters add detail
- Dolphin turn BACKWARD 0.125 rotations add detail
- Dolphin move FORWARD 2.5 meters add detail
- Dolphin turn FORWARD 0.125 rotations add detail

Play Example

*"Okay, make the pig wave both hands up and both hands down to signal the helicopter."*

# Focus area labels



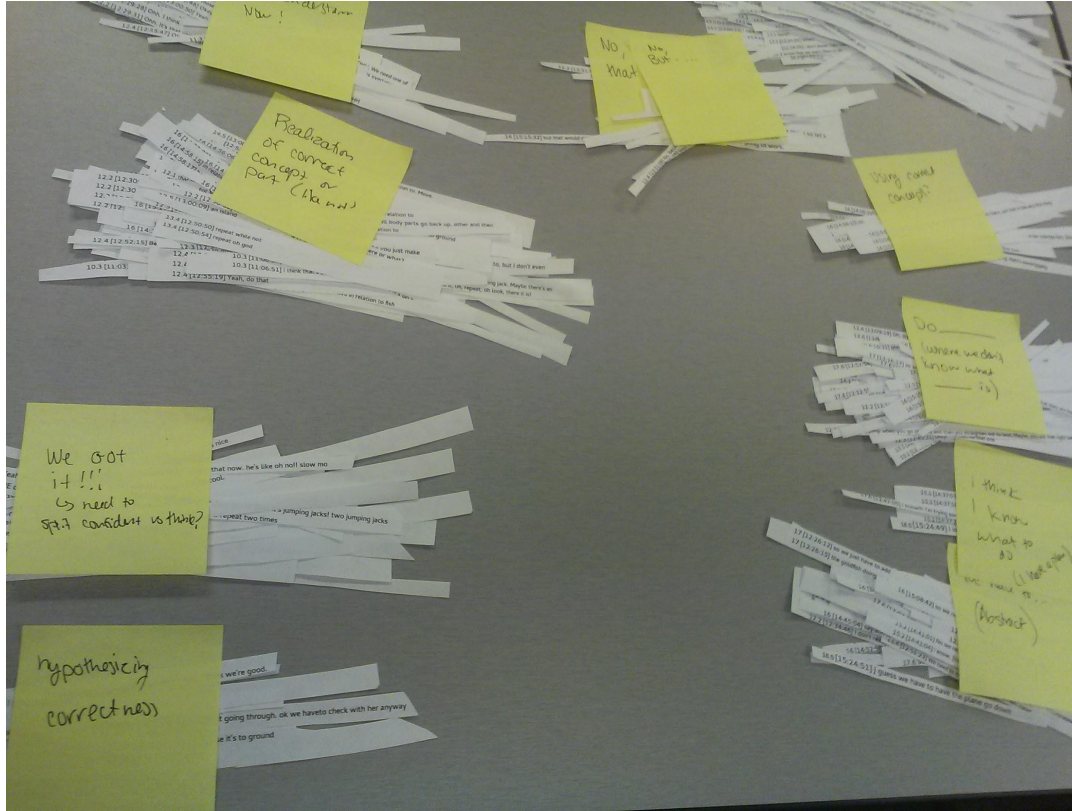
Task or Example  
Execution

*"Play it."*

# Focus area labels

Unknown/Other	<i>"That was hard!!!"</i>
Off-topic	<i>"Eat fish while the eat fish eat a fish."</i>

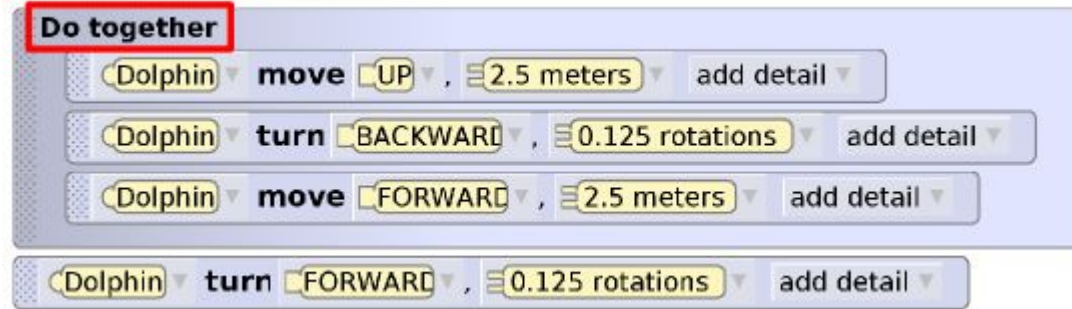
# Qualitative Analysis: Transcription Labeling



## 7.6 hours of audio

# Focus area labels

Example Code: *"It says do together, so we should try to find a way to do that."*



# Existing Example Support

Sample Code in the inline CSS format:

Save this code for later use

```
<A href='your_url'  
style='color: rgb(53, 159, 0);  
font-family:  
Arial,Helvetica,sans-serif;  
font-size: 13px; text-align:  
justify; '>Your text.</A>
```

Sample Code in the separate CSS format:

Save this code for later use

```
/*css*/  
.your_class {  
color: rgb(53, 159, 0);  
font-family:  
Arial,Helvetica,sans-serif;  
font-size: 13px;  
text-align: justify;  
}  
/*html*/  
<A href='your_url'  
class='your_class'>Your  
text.</a>
```

# Process labels

**Unknown/other**

***“What a beautiful pig.”***



# Code Misconception Hurdle

“Maybe you put the right shoulder, maybe you switch those around. [...] Cause then it would go in sync.”

# Code Misconception Hurdle

“Maybe you put the right shoulder, maybe you switch those around. [...] Cause then it would go in sync.”

## Task Code

```

Pig say "I'm here!! I'm here!!" add detail
Pig getRightShoulder turn BACKWARD, 0.25 rotations add detail
Pig getRightShoulder turn FORWARD, 0.25 rotations add detail
Pig getLeftShoulder turn BACKWARD, 0.25 rotations add detail
Pig getLeftShoulder turn FORWARD, 0.25 rotations add detail

```

## Correct Solution

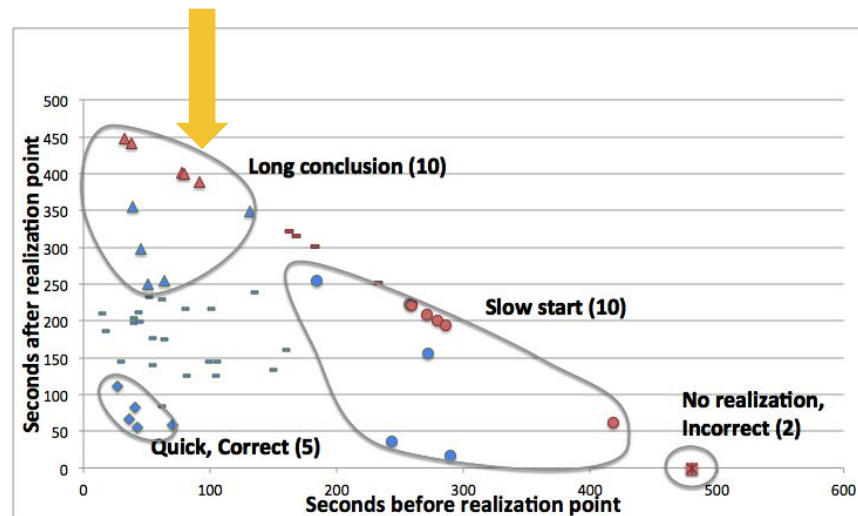
```

Pig say "I'm here!! I'm here!!" add detail
Do together
  Pig getRightShoulder turn BACKWARD, 0.25 rotations add detail
  Pig getLeftShoulder turn BACKWARD, 0.25 rotations add detail
Do together
  Pig getLeftShoulder turn FORWARD, 0.25 rotations add detail
  Pig getRightShoulder turn FORWARD, 0.25 rotations add detail

```

# Code Misconception Hurdle

Label	Long Conclusion, incorrect
Task Code: idea	3 ✖ average (A)
Task Code Execution	2.5 ✖ average (A)

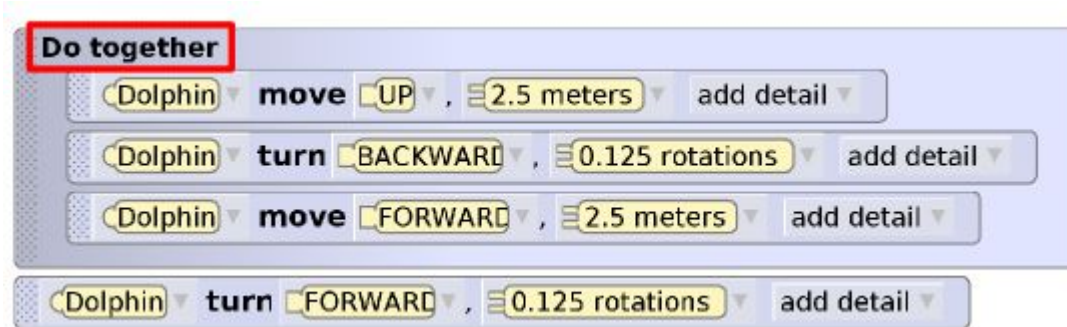


# Example Emphasis Strategy

“We just saw the outline.”

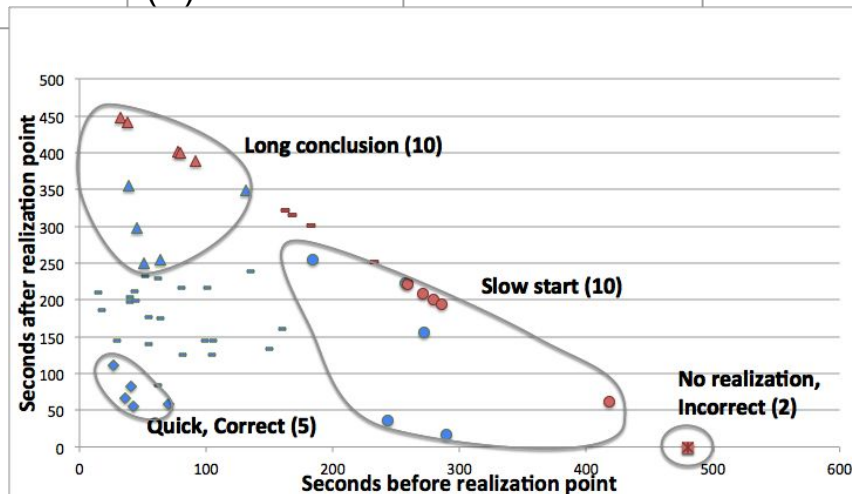
# Example Emphasis Strategy

“We just saw the outline.”



# Example Emphasis Strategy

Label	Slow Start, correct	Slow Start, correct	Quick, correct	Long Conclusion, correct	Long Conclusion, incorrect
Example Code	1.7 ✖ avg. (B)	1.7 ✖ avg. (B)	0.3 ✖ avg. (B)	1.3 ✖ avg. (B)	1 ✖ avg. (B)



# Example use as part of non-expert programming behavior

- Code Reuse<sup>1</sup>
- Debugging<sup>2</sup>
- Programming behaviors<sup>3</sup>
- **[Programming with examples]** ★

1. P. Gross and C. Kelleher, "Non-programmers identifying functionality in unfamiliar code: strategies and barriers," J. Vis. Lang. Comput., vol. 21, no. 5, pp. 263–276, 2010. 2.3, 1996. C. Kissinger, M. Burnett, S. Stumpf, N. Subrahmaniyan, L. Beckwith, S. Yang, and M. B. Rosson, "Supporting end-user debugging: what do users want to know?," in Proceedings of the working conference on Advanced visual interfaces, 2006, pp. 135–142.

3. A. J. Ko, B. A. Myers, and H. H. Aung, "Six learning barriers in end-user programming systems," in Visual Languages and Human Centric Computing, 2004 IEEE Symposium on, 2004, pp. 199–206.