Code Puzzle Completion Problems in Support of Learning Programming Independently

Kyle J. Harms



Lots of Jobs... Too Few Computer Science Graduates



8% of STEM graduates are in computer science

2

K-12: Lack of Access to Learn Programming



2

"High schools are 5 times more likely to have a football team than a computer science program."



Kurose, Jim. "An Expanding and Expansive View of Computing". 2016. Presentation. https://engineering.wustl.edu/Events/Pages/CSE-Colloquia-Series-Jim-Kurose.aspx

How can we help children learn programming independently?

Massive Open Online Courses

Subjects ▼ Q	KHANACADEMY	New user
 INTRO TO JS: DRAWING & ANIMATION Drawing basics 	E Making drawings with code We'll explain how to draw circles with code (JavaScript and ProcessingJS), and then you'll get to try it yourself in a challenge.	s
Making drawings with code Quick tip: number scrubbing Challenge: Simple snowman Drawing more shapes with code Challenge: Waving snowman Challenge: Waving snowman	<pre>1 ellipse(212, 206, 283, 318); 2 ellipse(212, 250, 100, 73); 4 ellipse(150, 150, 30, 30); 6 7</pre>	

Motivating Novice Programming Environments



lookingglass.wustl.edu

Programming Tutorials



scratch.mit.edu

Code Puzzles



Cognitive Load Theory

J. Sweller and P. Chandler, "Why Some Material Is Difficult to Learn," Cognition and Instruction, vol. 12, no. 3, pp. 185–233, 1994. Sweller, P. Ayres, and S. Kalyuga, Cognitive Load Theory. Springer, 2011.

Human Cognitive Architecture





Working Memory

Long Term Memory

10 G. A. Miller, "The magical number seven, plus or minus two: some limits on our capacity for processing information," Psychological Review, vol. 63, no. 2, pp. 81–97, 1956.

Cognitive Load

Beneficial

Intrinsic Cognitive Load

Germane Cognitive Load

Detrimental

Extraneous Cognitive Load







Working Memory

11 Sweller, P. Ayres, and S. Kalyuga, Cognitive Load Theory. Springer, 2011.

Beneficial Cognitive Load

Write a program that sorts an array of integers.

```
public static void bubbleSort( int[] elements ) {
    int tmp;
    for( int i = 0; i < ( elements.length - 1 ); i++ ) {
        for( int j = 1; j < ( elements.length - i ); j++ ) {
            if( elements[ j - 1 ] > elements[ j ] ) {
                tmp = elements[ j - 1 ];
                elements[ j - 1 ] = elements[ j ];
                elements[ j ] = tmp;
            }
        }
}
```

12

Low Beneficial Cognitive Load

Write a program that sorts an array of integers. When you are finished, comment your code!

```
public static void bubbleSort( int[] elements ) {
    int tmp; // swap variable
```

```
// check all elements in the array
for( int i = 0; i < ( elements.length - 1 ); i++ ) {
    // move the next largest element to the end
    for( int j = 1; j < ( elements.length - i ); j++ ) {
        // compare adjacent elements; swap elements if
        // they are in reverse order
        if( elements[ j - 1 ] > elements[ j ] ) {
            // swap the elements
            tmp = elements[ j - 1 ];
        }
    }
}
```

```
elements[ j - 1 ] = elements[ j ];
elements[ j ] = tmp;
```

High Beneficial Cognitive Load

Detrimental Cognitive Load

Programming Syntax

```
public static int sum( int min, int max ) {
    assert max > min;
    int sum = 0:
    for( int i = min; i <= max; i++ ) {
        sum = sum + i;
    }
    return sum;</pre>
```

Blocks-Based Programming





13



Low Detrimental Cognitive Load

Completion Problem

14

"A partial worked example where the learner has to complete some key solution steps."



J. J. G. Van Merrienboer and M. B. M. De Croock, "Strategies for Computer-Based Programming Instruction: Program Completion Vs. Program Generation," Journal of Educational Computing Research, vol. 8, no. 3, pp. 365–394, Jan. 1992. Sweller, P. Ayres, and S. Kalyuga, Cognitive Load Theory. Springer, 2011.

Example Problem

What is ∠BCA?



Conventional vs. Completion Problems

Conventional Problem

Completion Problem

What is $\angle BCA$?

What is $\angle BCA$?

∠BCA = ____

 $\angle BAC + \angle CBA + \angle BCA = 180^{\circ}$ 30° + ____ + $\angle BCA = 180^{\circ}$





Completion Problems: Beneficial Cognitive Load

Conventional Problem

Completion Problem

What is \angle BCA?

 $\angle BCA = 60^{\circ}$

What is $\angle BCA$?

$$\angle BAC + \angle CBA + \angle BCA = 180^{\circ}$$
$$30^{\circ} + \frac{90^{\circ}}{2} + \angle BCA = 180^{\circ}$$
$$\angle BCA = 180^{\circ} - 30^{\circ} - 90^{\circ}$$
$$\angle BCA = 60^{\circ}$$





Motivating Programming Environment



LBLA = 60°

How might we combine a motivating programming environment, code puzzles, and completion problems?

Code Puzzle Completion Problems

Yeti Baseball – Looking Glass		×
Play Correct Play Mine tiny yeti jump to hammer tiny yeti fly away Do together big yeti big yeti big yeti big yeti big yeti	Undo Redo Reset X Quit The animation, Yeti Baseball, is all mixed up! Using only the pieces you need, put the animation back in the correct order. (try yet) turn to face big yet) Do together big yet1 nod (try yet] move BACKWARD 0.5 meters	

Do code puzzle completion problems support children in using programming constructs in near transfer situations independently?

Hypotheses

Hypothesis I Middle school children will demonstrate greater learning efficiency of programming constructs by finishing training materials in less time and successfully completing more near transfer tasks, when training independently using code puzzle completion problems compared to current best practice, tutorials.

Hypothesis II Errors in code puzzles, commonly called distractors, will decrease learning efficiency by increasing middle school children's time to complete code puzzle completion problems while also decreasing their ability to successfully utilize programming constructs in near transfer situations.

Hypothesis III When given the freedom to choose their own informal learning resources, middle school children will perceive value in using code puzzle completion problems by expressing more interest, enjoyment, and preference towards using them compared to tutorials.

Formative Evaluations: Developing Code Puzzle Completion Problems & an Introductory Curriculum

K. J. Harms, N. Rowlett, and C. Kelleher. 2015. Enabling independent learning of programming concepts through programming completion puzzles. In 2015 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), 271–279. https://doi.org/10.1109/VLHCC.2015.7357226

Formative Evaluations

I. Completion Problem → Puzzle Interface & Mechanics

10 iterations

23 participants - St. Louis Science Center

30 minutes

II. Introductory Curriculum

8 iterations

21 participants - St. Louis Academy of Science

90 minutes

Things we tried:

Scrambling the code.

Fun & unpredictable animations.

Feedback only after solving the puzzle.

Do other activities while watching program execution.

Overlapping puzzle statements.

Demonstrating construct behavior/example-based puzzles.

Detailed code and animations.

Scrambling the code.

What is a Programming Completion Problem?

Give the first few lines?

Leave blanks throughout the code? Scramble the order of the statements?



Code Scramble



Code Scramble: Many Editable Dimensions





Limit the Editable Dimensions



Fun & unpredictable animations.

Fun & Unpredictable: Hard to Remember



Brown Monkey turn to face	Grey Monkey			
Grey Monkey) turn to face	Brown Monkey			
Repeat 2 times				
Brown Monkey talk				
Grey Monkey talk				
[⊥] loop [⊥]	ļ			
Brown Monkey move UP	0.5 meters) duration			
Brown Monkey move DOW	N 0.5 meters duration			
Repeat 3 times				
Grey Monkey say "EEK EEK"				
Brown Monkey turn RIGHT 4.0 rotations				
Grey Monkey hop side t	o side			
loop				
Grey Monkey] lay down				



Memorable Segments



Segment III - alien leaves in flying saucer

Segment I – alien repairs flying saucer

Segment II – flying saucer starts up



Memorable Segments



Segment I	alien walk 0.5 meters Repeat 2 times alien kick flying saucer shake loop
Segment II	Do together flying saucer fly above alien flying saucer turn RIGHT 2.0 rotations duration 2.0 seconds
Segment III	flying saucer beam up alien flying saucer fly away

Motivating Programming Environment + Code Puzzle Activity + Completion Problems
Code Puzzle Completion Problems

*al	lien-and-ufo.lgp - Looking Glass	×
Play Correct Play Mine Use all of these actions to put the animation back in the correct order. (flying saucer turn RIGHT 2.0 rotations du Repeat 2 times Ioop (flying saucer fly above alien alien walk 0.5 meters	custom action My Story	Done 💥
alien walk 0.5 meters		

Introductory Curriculum



1. Sequential



4. Repeated & Parallel



2. Repeated



5. Parallel { Repeated }



3. Parallel



6. Repeated { Parallel }

Hypotheses

Hypothesis I Middle school children will demonstrate greater learning efficiency of programming constructs by finishing training materials in less time and successfully completing more near transfer tasks, when training independently using code puzzle completion problems compared to current best practice, tutorials.

Hypothesis II Errors in code puzzles, commonly called distractors, will decrease learning efficiency by increasing middle school children's time to complete code puzzle completion problems while also decreasing their ability to successfully utilize programming constructs in near transfer situations.

Hypothesis III When given the freedom to choose their own informal learning resources, middle school children will perceive value in using code puzzle completion problems by expressing more interest, enjoyment, and preference towards using them compared to tutorials.

Summative Evaluation: Hypothesis I – Are code puzzle completion problems effective?

K. J. Harms, N. Rowlett, and C. Kelleher. 2015. Enabling independent learning of programming concepts through programming completion puzzles. In 2015 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), 271–279. https://doi.org/10.1109/VLHCC.2015.7357226

Literary Search: Tutorials

Scratch Project Editor – Imagine, Program, Share – Google Chrome	🛞 🏫 All Tips
Scratch Project Edite ×	🚨 < васк 1 of 13 нехт >
😵 🗞 🔮 https://scratch.mit.edu/projects/editor/?tip_bar=getStarted	
🔐 File 🔻 Edit 🕶 Tips About 🕹 👫 💥 🍘 Sign in to save Sig	Start Moving
Untitled Costumes Sounds	N
X: 240 Y: -177 Sprites New sprite: Y: 20 Y: -177 Witch backdrop to lockdrop! Sprites New sprite: New sprite: Sprites New sprite: Sprite Sprite Sprite Sprite New backdrop: Sprite Sprite Sprite Sprite Sprite Sprite Sprite	<text><text><text><text></text></text></text></text>

"Scratch," Scratch. [Online]. Available: https://scratch.mit.edu/.

K. J. Harms, D. Cosgrove, S. Gray, and C. Kelleher, "Automatically Generating Tutorials to Enable Middle School Children to Learn Programming Independently," 2013.

Hypothesis I

Middle school children will demonstrate greater learning efficiency of programming constructs by finishing training materials in less time and successfully completing more near transfer tasks, when training independently using code puzzle completion problems compared to current best practice, tutorials.

Tutorials

tw	o-walruses.lgp – Looking Glass (1)
World Edit Help	
🚱 Create or Open Project 🕌 Save 🦘 Undo 🇬 Red	io 🕨 Play
	Scene My Story custom action My Story Drop action here.
baby walrus's Actions	
new action	0
Custom Actions (2)	
edit baby walrus shake head	
edit baby walrus turn back to normal color	
say, think	
() (baby walrus) say (text)	
baby walrus think text	
position	
(?) baby walrus move (direction) (amount)	
baby wairus move toward target amount	
baby wairus move to target	
haby walrus place spatial relation (target)	
	l

Tutorial •		
Baby walrus say weeeeeeee!		
Scene MyStory Custom action My Story Custom action My Story Custom action My Story Custom action here. Frep action here.		
Drag in the following action: baby walrus say "weeeeeeeee!".		
Next »		

Summative Evaluation

Between-Subjects Study

Control: Tutorials

Experimental: Puzzles

27 participants (10 to 15 years)

12 Female, 15 Male

Average Age: 11.59

Minimal Programming Experience (< 3 hours)

2 hours

Study Design





Study Design





Training Tasks × 6



Control: Tutorials

Experimental: Puzzles

Study Design



Training Phase



Transfer Tasks × 4



Correct Animation

Completed Transfer Task



Initial State

Completed Task

Hypothesis I

Middle school children will **demonstrate greater learning efficiency** of programming constructs by finishing training materials in less time and **successfully completing more near transfer tasks**, when training independently using code puzzle completion problems compared to current best practice, tutorials.



Tutorials Take Longer To Complete Than Puzzles

Mean Training Task Completion Time Across Training Tasks



 $V = .75, F(6, 29) = 14.12, p < .001, \eta^2 = .75$

Puzzles Impose More Cognitive Load Than Tutorials

Mean Training Task Cognitive Load



 $V = .51, F(6, 29) = 4.45, p = .003, \eta^2 = .51$

Transfer Task Performance

Puzzle Users Performed Better on Near Transfer Tasks

Mean Transfer Task Performance



Hypothesis I: Accept



Code puzzle completion problem participants performed 33% better on transfer tasks while requiring 21% less training time compared to tutorial participants.

Summative Evaluation: Hypothesis II – Do errors in code puzzles decrease learning?

Kyle James Harms, Jason Chen, and Caitlin L. Kelleher. 2016. Distractors in Parsons Problems Decrease Learning Efficiency for Young Novice Programmers. In Proceedings of the 2016 ACM Conference on International Computing Education Research (ICER '16), 241–250. https://doi.org/10.1145/2960310.2960314

Parsons Problems & Distractors

csp-8-3-4: The following is the correct code for printing the even numbers from 0 to 10, **but it also includes some extra code that you won't need**. Drag the needed blocks from the left and put them in the correct order on the right. Don't forget to indent blocks in the body of the loop. Just drag the block to the further right to indent. Click the *Check Me* button to check your solution.



Parsons, D. and Haden, P. 2006. Parson's Programming Puzzles: A Fun and Effective Learning Tool for First Programming Courses. Ericson, B.J. et al. 2015. Analysis of Interactive Features Designed to Enhance Learning in an Ebook.

Errors and Learning

Learning with Errors: Teach Common Misconceptions

Print TRUE if string1 and string2 contain the same value.

```
String string1 = new String("abc");
String string2 = new String("abc");
if (string1 = string2) {
   System.out.println("TRUE");
} else {
   System.out.println("FALSE");
} String string1 = new String("abc");
String string2 = new String("abc");
if (string1.equals(string2)) {
   System.out.println("TRUE");
} else {
   System.out.println("FALSE");
} String string1 = new String("abc");
} String string2 = new String2 = ne
```

Muller, D. a. et al. 2008. Saying the wrong thing: improving learning with multimedia by including misconceptions. Große, C.S. and Renkl, A. 2007. Finding and fixing errors in worked examples: Can this foster learning outcomes? Wang, M. et al. 2015. Using Feedback to Improve Learning: Differentiating between Correct and Erroneous Examples.

Distractors in Multiple Choice Tests

1. Consider the following code segment.

```
String string1 = new String("abc");
String string2 = new String("abc");
if (string1 = string2) {
   System.out.println("TRUE");
} else {
   System.out.println("FALSE");
}
```

What is printed as a result of executing the code segment?



Richland, L.E. et al. 2009. The pretesting effect: Do unsuccessful retrieval attempts enhance learning? Little, J.L. and Bjork, E.L. 2012. The persisting benefits of using multiple-choice tests as learning events. Roediger III, H.L. and Marsh, E.J. 2005. The Positive and Negative Consequences of Multiple-Choice Testing.

Testing & Information Retrieval

1. Consider the following code segment.

```
String string1 = new String("abc");
String string2 = new String("abc");
```

```
if (string1 = string2) {
   System.out.println("TRUE");
} else {
   System.out.println("FALSE");
}
```

What is printed as a result of executing the code segment? (a) TRUE (b) FALSE (c) TRUE FALSE

Richland, L.E. et al. 2009. The pretesting effect: Do unsuccessful retrieval attempts enhance learning? Little, J.L. and Bjork, E.L. 2012. The persisting benefits of using multiple-choice tests as learning events. Roediger III, H.L. and Marsh, E.J. 2005. The Positive and Negative Consequences of Multiple-Choice Testing.

The Testing Effect

1. Consider the following code segment.

```
String string1 = new String("abc");
 String string2 = new String("abc");
 if (string1 = string2) {
   System.out.println("TRUE");
  } else {
   System.out.println("FALSE");
What is printed as a result of executing the code segment?
               (b) FALSE
(a) TRUE
                              (c) TRUE
                                 FALSE
```

Bridger, E.K. and Mecklinger, A. 2014. Errorful and errorless learning: The impact of cue–target constraint in learning from errors. Roediger, H.L. and Karpicke, J.D. 2006. Test-Enhanced Learning Taking Memory Tests Improves Long-Term Retention. Karpicke, J.D. 2012. Retrieval-Based Learning Active Retrieval Promotes Meaningful Learning.

A Distractor is an *Error*...



Hypothesis II

Errors in code puzzles, commonly called distractors, will decrease learning efficiency by increasing middle school children's time to complete code puzzle completion problems while also decreasing their ability to successfully utilize programming constructs in near transfer situations.

Formative Evaluation: What is a Distractor in a Code Puzzle?

"Unnecessary Code"

"Extra Fragments"

"Erroneous Code"

Distractors can be used to...

"illustrate a particular point"

"highlight programming principles the student may ignore"



Summative Evaluation

Between-Subjects Study

Control: No Distractors

Experimental: Distractors

92 participants (10 to 15 years)

32 female, 60 male

Average age: 12.9 years

2 hours

Study Design: Training Phase





Study Design: Training Phase



Transfer Phase



Training Tasks × 6



Control: No Distractors

Experimental: Distractors

Study Design: Transfer Phase

Training Phase Familiarization Task Training Task Task Survey ×6 CS CLCS Survey


Transfer Tasks × 3

World Edit Help Save Sure Redo Play X Done Scene Hungry Monkey custom action Hungry Monkey	Task Instructions X
Le Save → Undo ← Redo Scene Hungry Monkey custom action Hungry Monkey	Task Instructions X
Scene Hungry Monkey custom action Hungry Monkey	Task Instructions X
custom action Hungry Monkey	Task Instructions X
 Try your best to m All of the existing You may only add 	nake your animation match the correct animation below. actions are already in the correct order. d up to 3 additional actions or action ordering boxes total .
 Coconut (6) coconut (6) straighten out joints Coconut (6) straighten out joints 	
Action Ordering Boxes	
Do in order	
Do actions at the same time Do together	
Repeat actions multiple times Repeat number times loop Repeat while condition loop	



Hypothesis II

Errors in code puzzles, commonly called distractors, will **decrease learning efficiency** by increasing middle school children's time to complete code puzzle completion problems while also **decreasing their ability to successfully utilize programming constructs in near transfer situations**.



Distractors Increased Puzzle Completion Time

Mean Training Task Completion Time



 $V = .17, F(6, 81) = 2.86, p = .014, \eta^2 = .17$

Distractor Participants Completed Fewer Puzzles Successfully

Percentage of Participants Who Successfully Completed Each Training Task



 $V = .30, F(6, 81) = 5.65, p < .001, \eta^2 = .30$

Distractors Increased Cognitive Load (Extraneous) $F(1, 86) = 6.03, p = .016, \omega^2 = .05$

Mean Cognitive Load Ratings for Each Training Task



Transfer Task Performance

No Difference in Transfer Task Performance

Percent of Correctly Completed Transfer Tasks



80

Hypothesis II: Partially Accept



Distractors increased learners' cognitive load, increased their time on task by 14% and decreased their success at completing code puzzles by 26%.

However, distractors did not significantly effect transfer task performance.

Summative Evaluation: Hypothesis III: Will learners be motivated to choose to use puzzles?

K. J. Harms, E. Balzuweit, J. Chen, and C. Kelleher. 2016. Learning programming from tutorials and code puzzles: Children's perceptions of value. In 2016 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), 59–67. https://doi.org/10.1109/VLHCC.2016.7739665

Goal-Oriented Learners





Independent Learning Support in Novice Programming Environments







Hypothesis III

When given the freedom to choose their own informal learning resources, middle school children will perceive value in using code puzzle completion problems by expressing more interest, enjoyment, and preference towards using them compared to tutorials.

Summative Evaluation

30 participants (10 to 15 years)

14 female, 16 male

Average age: 11.2 years

2 hours

Instructional Tasks



Dizzy Walrus



Hammer Hazard



Monkey Business



Yeti Baseball



The Snow Dance



Messed Up Magic



Space Mechanic



Trouble at Sea



Whack-a-Yeti



Crazy Cauldron





Firetruck Frenzy



Air Traffic



Polar Surprise



Choose Instructional Task × 6



As Tutorial or Puzzle...



Semi-Structured Interviews

Pre-Study Interview

"How would you rate your current programming or coding expertise? why?"

Post-Task Interviews × 6

"Was the experience of completing the tutorial/puzzle valuable to you in any way? why?" "Did you learn anything new or did you acquire any new skills while doing the tutorial/puzzle?"

Post-Study Interview

"When is it better to to use tutorials/puzzles on your own? why?"

•••

...

...

Interview Response Analysis: Categories

J. Stranger Manager	make a part	13 41 41 Annual Production of the first program for the pr
2.6 67 4 puttle present they do you par that? Performance Decards Reproduce and where everything goes and what to put in the do together here and what goes out of the hos.	2.32 e3 e to	14 07 C Vertexineer: Why would go the point expertise routs compared? Purtigent: Notices I benefit a lart block coding and have it and the durations and everything. Purtigent: Yes.
22 44 2 Interval Participant; Inst. It's easier than a lot of the things //is programmed with	Participant: Protody a four Heritigant: Protody a four Heritigant: Stranged II have been a four as a fournut?	2.2 45 6 bitotil Headware Ca kry on Ut fire entry? Personal cases and the personal register of the case of the personal register of the personal cases and the cases and t
2.14 44 6 based and the second	2.20 12 2 particle This finance in the low of end in the low of e	11 44 3 Burlow Research from the que on approximation to a to to toget approximation and approximation to the query toget approximation toget approximatis toget approximation toget approximatis toget approximation toget
osher crea, but it was prote simple actually.	Participant: English have municipant for for most indication mataxies of something? Manimum ry to might have municipant for most a mistake. Participant: Yes.	
2.12 EF 5 Definition of the second determined and the second determine	14 41 2 pcccl Performant Mode to Store or or two. Mannessee 16 and pcsce and pcsce and pcsce 14 41 2 pccc Performant Mode to Store or or two. Performant Mode to Store or or two. montessee montessee pccc Performant Mode to Store or or two. montessee montessee pccc Performant Mode to Store or two. montessee montessee pccc Performant Mode to Store or two. montessee montessee pccc Performant Mode to Store or two. montessee montessee pccc Performant Mode to Store or two. montessee montessee pccc Performant Mode to Store or two. montessee pccc pccc pccc Performant Mode to Store or two. montessee pccc pccc pccc pccc	
2.4 S2 S puzzle Percelogate Produkty, the moderately to very difficult Percelogate Produktion the enold you pick? Percelogate Produktion Percelogate Percelogate Produktion Percelogate P	5.4 64 7 public Merchant Three. So, why do you say that, a three, for a storytul? Memchant Alberter, built give weighting and percentage meet to do.	
Participant: Decavity in our to the brank. I was going to be very difficult? 2.10 39 6 pacety Internativeser: I think it would have been extremely even been set	3.3 44 X Participant: Yeak, becase I learned how to just plan X with the puzzle, because you watch it. Then you plan out what happens, and there is the tutorial, you just [gare out what to do and how to do it. So when you	214 30 7 public Participant: Recruite Its blink progla danced. 214 50 7 public Participant: Recruit Recruit Its blink progla danced.
Participant: The annualize locks a title more complicated, I share I would need the future annualizes	K.S.1 39 8 Participant Birl Holic in the Easter. K.S.1 39 8 Participant Birl Holic in the photole use possiles for things that even't so easy and tuterlais for things that they binds are exampled.	a logant
4.3 35 2 extensioners that may all advances and a data international framework is a data with the second of the second as a data with the second as	3.3 37 E Methogenet Prototoly when you don't low-what r going don, you don't follow the fuderal, and when you're blacked and an experiment of the state of a state of the	Line Vision
2.8 73 4 2.8 73 4 2.8 2.8 2.8 2.8 2.9 2.8 2.9 2.8 2.9	gen a une possente en or y a. And San	X.7.1 S1 9 Participant: I think Shark Shark Security is a list more complicated and I liked the finished product. Participant: Yush
Perticipant: trait. Weil, no. 2.12 73 7 Tutorial Participant: Instability put because, at certain parts, it was a for mechanic	needs to inclusion and possible. Doubling accountly do it when the pretry good with the program and () ust Performance in the provide the provide the provide the provide the provide the program and () ust Performance in the provide the providet the provide the provide the providet the pr	22 37 6 tetroite Convoit free who
Perticipant de l'Andrea tatorial de valt, bis est a tootial, report Manufaceur teat, rou de de autorial and rou de toaterial.	a.s. s s2 / Lidenial water to figure none of a local on my own. Additional figure none of a local on my own. Additional figure none of a local on my own. Additional figure none of a local on my own.	2.34 63 3 puttin Participant (3) reak. (4) it because it van an eavy one but it waart take one of the super easy one, plus it built in of footierd fun.
https://device.it/jii/site/u. https://device.it/jii/site/u. Autholpanet_40; Youh	Participant. All the reason that I gave	2.1 56 4 puede "Production or weapent. Perfoldant: Certing to do 9. Usad hour late hou
3.53 35 P Interlogues 35, including combine 38. 36 P Interlogues 45, including the Altradic. Participant 55, tro, including it was complete built a water 1 basis.	As 4 Professional and points As a professional and points As a professional and points As a professional As means the institute and points and points into the basics about all the safe, i which the table reads to learn the skills again, und to the basics which probably be more processing on my examines and points and point	22 S0 4 pupels 24 and how is all cares tagetter at the end. 25 S0 4 pupels 26 and pupels 27 and pupels 27 and pupels 28 and pupels 28 and pupels 29 and pupels 20 and pup
	3.3 46 7 Participant Tatorica, https://www.tatorica.com/	Perfolgant 2 voir velly test) 23 39 2 profe Perfolgant 39 real.
	Participant secure of the processing state Participant EL (inspire) Participant EL (inspire) Participant EL (inspire)	23.2 23 4 pucle Participant 3R There's noting bar about it. Participant Vesh. Participant Vesh.
	Participant 61 Just because like the name tary, fi's a public and you have to figure it out yourself and it's more of a coulterge instead of Just following the instructions and the instructions a meteriorement 5 is a challinger a good thing or is but a lade thing?	2.2 62 7 puzzle Meteriever What about it did you enjoy? Participant: enjoyed, 1 don't how. I jout enjoyed it. It was fun to figure out.
the second se	Partogant is 1. for 12 a good twid, Interviewer Sy you want to be challenged? Participant 61: Yeah.	2.1 58 3 pumbe Interviewer: What and it fun? Participant: Decuse it was challenging. It previded a challenge and i sought the challenge.
	Archigant: Lenjoy puzzles. Si Archigant: Lenjoy puzzles free with you only the puzzles more than the totarials? Archigant: Elite Ankalenge of the puzzles, more as than the totarials.	2.14 47 4 puzzle Perificipant Encourse): Na monthly: Perificipant Juni Decours I har monthly: Perificipant Juni Decours I har monthly:
9 Participant: Maybe next year when i'm in that class, maybe I should learn something them. 1.1 57 8 Interviewer: Glau, 50 what can that club give you that tutorials and perifer can't?		2.14 64 3 Exterial Participant it seemed interesting.
Participant: Probably more experience, there glocates with the computer and of programming. Participant: Locating class could get a could get an and Scratch because for acch is trickler, but Locating Gass could be voir.	1.0 40 5 Participant 40 to https://	
44 a Patentiewer Se what is it about Social when costs plate is to make it in rolely hand and make you work more Participant: week in Socially it's taking, bit you have a figure it put to the charaka and the putole cannet? Figure 1 out, but it's cannot put to social the social is you have a figure it put to it is conting Gase, it's lived at law you	2.8 40 4 puzzle Participant 440: (thought it was going to be a six, but then, it turned out to be a flow for me. L	
too have to make the characters. You got to do every single small thing which is harder.	3.8.2 68 5 Participant: A Role. 3.8.3 37.4 8 Participant: Number two.	Participant 35: Weil, it give me a serve of accomplishment.
Participant: Just practice probaby.	2.7 73 4 puttle Participant No.	2.5 55 2 puzzle Interniewen Cikay: That's good. Anything ene? Participant 55: Just that I could waith scientifing and then complete it.
1.1 59 51 Interviewer: Woold you practice with? 1.1 59 51 Interviewer: Woold you gate and the coding program. 1.1 59 51 Interviewer: Woold you dowhat would you do?	2.16 54 4 puzzle Participant: (dor') know 8.64 56 4 Participant: (dor') know	2.3.1 20 8 puzzle Participant: Figuring this out. Multi-All 3.3.1 40 2 puzzle Participant: Watching my animation when it was done correctly. Multi-All
percesses we become and publicly of percession and rerun through them with publics and totopiats.	17 46 6 pourle Participant 61 No.	24 61 6 punie Perticipant 61, Not really.
	2.3.2 S4 S Interval Participants rearit ready more of anything	2.4 39 6 puzzle Interviewent Not really 2.5 45 5 puzzle Participant Well (don't bink to:
	3.8.1 45 8 Participant: The Whark a test, I thek.	Pertidipant: Mesord up Magin:
	Andráve Char pre event ha film Andráve Char pre predicate. Andráve Martí Para pre event ha film Andráve Martí Para pre event ha film Andráve Martí Para pre event ha film	3.7.3 III 5
	Participante Auto and management of the set	2.11 73 6 purple think 1 might have even dow R, done the wrong thing over and over again. So, then it was rewarding when it finally worked.

High-Level Category Themes – Interrator Agreement

	Cohen's κ	р
High-Level Categories	.95	p < .001
Decision Rationales	.89	p < .001
Experience Outcomes	.86	p < .001
Expected Task Difficulty	.85	p < .001
Sources of Ease & Difficulty	.88	p < .001

Hypothesis III

When given the freedom to choose their own informal learning resources, middle school children will perceive value in using code puzzle completion problems by expressing **more interest, enjoyment, and preference towards using them compared to tutorials**.



Decisions: Preferred *X*

Instructional Format Decision Preference



Decisions: Primarily^{*} X

Instructional Format Primary Decision



*(\geq n - 1)

Preference vs. Primary Use

Instructional Format Decision Preference



Instructional Format Primary Decision



Puzzle Preference

Instructional Format Decision Preference

73% 60% 30% 10% 3% More Tutorials More Puzzles **Primarily Tutorials Both Tutorials & Puzzles Equal Tutorials & Puzzles**

Instructional Format Primary Decision

23%

Primarily Puzzles

High-Level Categories

	Cohen's κ	р
High-Level Categories	.95	p < .001
Decision Rationales	.89	p<.001
Experience Outcomes	.86	p<.001
Expected Enjoyment		p<.001
Sources of Ease & Difficulty .88		p<.001

Experience Outcomes: Liked Instructional Format

"I like the puzzles. They've all been fun."



Liked Tutorial 17% of participants



Liked Puzzle 43% of participants

Post-Study Interview: Which do you enjoy more?

Post-Study Enjoyment Preference Response for Instructional Format



Puzzle Preference

Instructional Format Decision Preference

73% 60% 30% 10% 3% More Tutorials More Puzzles **Primarily Tutorials Both Tutorials & Puzzles Equal Tutorials & Puzzles**

Instructional Format Primary Decision

23%

Primarily Puzzles

Using Tutorial and Puzzles

Instructional Format Decision Preference

Instructional Format Primary Decision



High-Level Categories

	Cohen's κ	р
High-Level Categories	.95	p < .001
Decision Rationales	.89	p<.001
Experience Outcomes	.86	p<.001
Expected Enjoyment	amming Skills	p<.001
Sources of Ease & Difficulty	.88	p<.001

Experience Outcomes: Improve Programming Skills



Tutorial Improved Programming Skills 60% of Participants



Puzzle Improved Programming Skills 70% of Participants

Using Tutorial and Puzzles

Instructional Format Decision Preference

Instructional Format Primary Decision



Decision Rationale Themes

		Cohen's ĸ	p
High-Level Cat	egories	.95	p<.001
Decisi	on Rationales	.89	p<.001
Experie	Personal Preference		p<.001
Expected	Challenge		p < .001
Sources of Ea	se & Difficulty	.88	p < .001
#1 Decision Rationale: Enjoyed Animation

"Probably just because it looked like a fun animation to do."



83% of participants made at least one decision because they enjoyed the animation.

Story-Based Animations are Motivating!

"I picked it as a tutorial because it looked like it had lots more complexity than the other ones and I didn't want to just jump right in without knowing what I was doing."



110 Caitlin Kelleher, Randy Pausch, and Sara Kiesler. 2007. Storytelling Alice Motivates Middle School Girls to Learn Computer Programming. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07), 1455–1464. https://doi.org/10.1145/1240624.1240844

Learners Like the Puzzles More, But Care More About the Animation

Instructional Format Decision Preference

111

Instructional Format Primary Decision



Caitlin Kelleher, Randy Pausch, and Sara Kiesler. 2007. Storytelling Alice Motivates Middle School Girls to Learn Computer Programming. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07), 1455–1464. https://doi.org/10.1145/1240624.1240844

Hypothesis III: Accept

- Participants preferred to choose puzzles over tutorials because they found the puzzles as more interesting, enjoyable, and challenging.
- However, participants often used both because they wanted to work on a specific animation.
- Participants also found both tutorials and puzzles useful for improving their programming skills.



Contributions

Formative Evaluations I & II:

Lessons learned from the development of code puzzle completion problems and curriculum.

Summative Evaluation: Hypothesis I

Code puzzle completion problems are efficient and effective in supporting independent learning.

Summative Evaluation: Hypothesis II

Errors in code puzzle completion problems decrease learning efficiency.

Summative Evaluation: Hypothesis III

Independent learners will choose to use code puzzles because they find them motivating and useful for improving their programming skills.



Future Work

Improving the Efficiency & Effectiveness of Code Puzzle Completion Problems

Using Worked Examples

Promoting Self-Explanation

Errors for Experienced Programmers

Extending Code Puzzle Completion Problems

Alternative Contexts

Curricular Content

Alternative Completion Approaches

Supporting Goal-Oriented Learners

Choices for Instructional Formats

Self-Directed Learning

Self-Assessment & Scaffolding

Completion Problems + Worked Examples

Yeti Baseball – Looking Glass	×
<pre>i Undo @ Redo @ Reset @ Out i Undo @ Indo @ Out i Undo @ Indo</pre>	 Example: The 'Do together' block makes these actions play at the same time. Do together penguin move UP, 10 penguin turn RIGH, 10 Worked Example Effect

Problem Completion *Effect*

John Sweller, Paul Ayres, and Slava Kalyuga. 2011. Cognitive Load Theory. Springer.

116 Michelle Ichinco, Kyle J. Harms, and Caitlin Kelleher. "Towards Understanding Successful Novice Example Use in Blocks-Based Programming". In: Journal of Visual Languages and Sentient Systems: Special Issue on Blocks Programming (2017). Forthcoming.

Programming Worked Examples

"A worked example provides a step-by-step solution to a problem."



Annotated Example?

Step 1:



Step 2:



Worked Example?



http://research.engineering.wustl.edu/~harmsk/

