Exploring the Impact of Worked Examples in a Novice Programming Environment

Rui Zhi
Thomas W. Price
Samiha Marwan
Alexandra Milliken
Tiffany Barnes
Min Chi
Introduction

Task 1: Calculate \((x+4)(3x-2)\)

\[
(x + 4)(3x - 2)
= x \cdot 3x + x \cdot (-2) + 4 \cdot (3x) + 4 \cdot (-2)
= 3x^2 - 2x + 12x - 8
= 3x^2 + 10x - 8
\]

A worked example for a math problem (Chen et al., 2018)
Introduction

- Worked examples have been studied in a variety of domains and can increase learning efficiency (Sweller et. al, 1985; McLaren et. al., 2014)

- However, only a few studies have compared worked examples to traditional problem solving in novice programming environments (Van Merriënboer & De Croock, 1992)
Cognitive Load Theory

- Cognitive Load Theory (Sweller et al., 1998)
Cognitive Load Theory

- Cognitive Load Theory (Sweller et al., 1998)
Cognitive Load Theory

Intrinsic

```java
int a;
a = 5;
```

vs.

```java
for (int i = 0; i < 5; i++) {
    ...
}
```

Cognitive Load
Cognitive Load Theory

Intrinsic

```java
int a;
a = 5;
```

vs.

```java
for (int i = 0; i < 5; i++) {
    ...
}
```

Extraneous

“A triangle is a polygon with three edges and three vertices.” - Wikipedia

Cognitive Load
Cognitive Load Theory

**Intrinsic**

```java
int a;
a = 5;
```

**Extraneous**

```java
for (int i = 0; i < 5; i++) {
    ...
}
```

---

"A triangle is a polygon with three edges and three vertices." - Wikipedia

---

Cognitive Load
Cognitive Load Theory

Intrinsic

```java
int a;
a = 5;
```

vs.

```java
for (int i = 0; i < 5; i++) {
    ...
}
```

Extraneous

“A triangle is a polygon with three edges and three vertices.” - Wikipedia

vs.

Germane

Cognitive Load

Rui Zhi, Thomas W. Price, Samiha Marwan, Alexandra Milliken, Tiffany Barnes, Min Chi
Cognitive Load Theory

Intrinsic
int a;
a = 5;

vs.
for (int i = 0; i < 5; i++) {
...
}

Extraneous

Germane

Cognitive Load

Rui Zhi, Thomas W. Price, Samiha Marwan, Alexandra Milliken, Tiffany Barnes, Min Chi
Cognitive Load Theory

- Cognitive Load Theory (Sweller et al., 1998)
Worked Examples

- Teaches problem-solving procedure by showing solutions step by step

\[
c (a + b) = \frac{af}{a} \\
\]
\[
c (a + b) = f \\
\]
\[
a + b = \frac{f}{c} \\
\]
\[
a = \frac{f}{c} - b \\
\]
\[
af + e = c \\
\]
\[
\frac{af + e}{b} = c \\
\]
\[
af + e = bc \\
\]
\[
af = bc - e \\
\]
\[
a = bc - e \\
\]
\[
f \\
\]

(Sweller & Cooper, 1985)

Rui Zhi, Thomas W. Price, Samiha Marwan, Alexandra Milliken, Tiffany Barnes, Min Chi
- Worked examples are one of the fundamental principles of programming education (Caspersen and Bennedsen, 2007)

- Suggest using worked examples in study materials and lectures (Vihavainen et al., 2011)

- Interleaving worked examples with practice problems can maximize students learning gains, compared to blocking WEs with problems, or solving equivalent problems (Trafton and Reiser, 1993)

- Incomplete worked examples improved novice's programming performance and post-test scores, compared with those who only had the WEs as a reference (MerrienBoer & Croock, 1992)

- It has been shown that combining self-explanation with WEs can be especially beneficial to students' learning (berthold, 2009)
Worked Examples in Programming

- Worked examples are one of the fundamental principles of programming education (Caspersen and Bennedsen, 2007)

- Suggest using worked examples in study materials and lectures (Vihavainen et al., 2011)

- Interleaving worked examples with practice problems can maximize students learning gains, compared to blocking WEs with problems, or solving equivalent problems (Trafton and Reiser, 1993)

- Incomplete worked examples improved novice's programming performance and post-test scores, compared with those who only had the WEs as a reference (MerrienBoer & Croock, 1992)

- It has been shown that combining self-explanation with WEs can be especially beneficial to students' learning (berthold, 2009)
Worked Examples in Programming

- Worked examples are one of the fundamental principles of programming education (Caspersen and Bennedsen, 2007)

- Suggest using worked examples in study materials and lectures (Vihavainen et al., 2011)

- Interleaving worked examples with practice problems can maximize students learning gains, compared to blocking WEs with problems, or solving equivalent problems (Trafton and Reiser, 1993)

- Incomplete worked examples improved novice's programming performance and post-test scores, compared with those who only had the WEs as a reference (MerrienBoer & Croock, 1992)

- It has been shown that combining self-explanation with WEs can be especially beneficial to students' learning (berthold, 2009)
Research Questions

How does having access to WErs during a programming problem impact:

- RQ1: Students’ learning during the problem?
- RQ2: Students’ perceived difficulty and cognitive load with respect to the problem?
- RQ3: Students’ programming efficiency?
Peer Code Helper

Chunk expert solution procedure into meaningful steps and present to students
Peer Code Helper

You will see how Ann made progress:

Ann's starting code:

The changes Ann made:

**Explanation:**
To modify the square (starting code) to draw a square. Ann changed the input of the block to increase the number of sides. She changed the inputs of the and blocks to make a polygon with sides so small it looks round.
Peer Code Helper

You will see how Ann made progress:

Ann's starting code:

Explanation: This code
Peer Code Helper

You will see how Ann made progress:

Ann's starting code:

```
Phase 1
Click me to run

go to x: 0 y: 0

clear

repeat 4

move 100 steps

pen down

turn 90 degrees

pen up

Explanation:
This code

Run Code

Submit
```

Visual Output
Peer Code Helper

You will see how Ann made progress:

Ann's starting code:

```
Phase 1
click me to run

Explanation:
This code draws a square shape.

go to x: 0  y: 0

repeat 4

move 100 steps

turn 90 degrees

pen up
```

Run Code
Submit

Rui Zhi, Thomas W. Price, Samiha Marwan, Alexandra Milliken, Tiffany Barnes, Min Chi
Peer Code Helper

You will see how Ann made progress:

Ann's starting code:

The changes Ann made:

Explanation:
To modify the square (starting code) to draw a
Ann changed the input of the
block to increase the number of sides. She
changed the inputs of the
blocks to make a polygon with sides so small it
looks round.
Peer Code Helper

You will see how Ann made progress:

Ann's starting code:

The changes Ann made:

Explanation:
To modify the square (starting code) to draw a circle, Ann changed the input of the *block* to increase the number of sides. She changed the inputs of the *block* and *block* to make a polygon with sides so small it looks round.

Rui Zhi, Thomas W. Price, Samiha Marwan, Alexandra Milliken, Tiffany Barnes, Min Chi
Peer Code Helper

You will see how Ann made progress:

Ann's starting code:  The changes Ann made:

Explanation:
To modify the square (starting code) to draw a
square, Ann changed the input of the block to increase the number of sides. She
cleaned up the code by removing unnecessary
blocks to make a polygon with sides so small it
looks round.
Peer Code Helper

**Phase 1**
Click me to run

go to x: 0 y: 0

clear

pen down

repeat 4

move 100 steps

turn 90 degrees

pen up

This code draws a square shape.

**Phase 2**
Click me to run

go to x: 0 y: 0

clear

pen down

repeat 36

move 9 steps

turn 10 degrees

pen up

To modify the square (Starting Code) to draw a circle, Ann changed the input of the repeat block to increase the number of sides. She changed the inputs of the move and turn blocks to make a polygon with sides so small it looks round.

**Phase 3**
Click me to run

go to x: 0 y: 0

clear

`Draw Circle`

Ann makes a new block with the code to draw a circle and gives it a name: "Draw Circle". This new block can be used many times. After "Phase 3" clicked, the new block is used one time.

**Phase 4**
Click me to run

go to x: 0 y: 0

clear

repeat 24

`Draw Circle`

turn 15 degrees

Ann makes the Draw Circle and turn block(s) happen 24 times to make a daisy design. When the code runs, it will draw a circle and then turn, 24 times.
Participants & Procedure

- **Participants**
  - 22 female high school students (ages 13 ~15)
  - Assigned to one of the two groups via matched pairs according to pre-test score

- **Two groups**

<table>
<thead>
<tr>
<th></th>
<th>E1</th>
<th>E2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem 1 (with WEs)</td>
<td>Problem 1 (without WEs)</td>
<td></td>
</tr>
<tr>
<td>Problem 2 (without WEs)</td>
<td>Problem 2 (with WEs)</td>
<td></td>
</tr>
</tbody>
</table>

**Problems**
- Problem 1: Daisy Design
- Problem 2: Spiral Polygon
- Problem 3: Brick Wall

Rui Zhi, Thomas W. Price, Samiha Marwan, Alexandra Milliken, Tiffany Barnes, Min Chi
# Study Outline

<table>
<thead>
<tr>
<th>Step</th>
<th>Group E1</th>
<th>Group E2</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Snap! Introduction (taught by camp instructor)</td>
<td></td>
<td>90 minutes</td>
</tr>
<tr>
<td>1</td>
<td>Experience pre-survey + <strong>Knowledge pre-test</strong></td>
<td></td>
<td>35 minutes</td>
</tr>
<tr>
<td>2</td>
<td>Introduce the Peer Code Helper</td>
<td></td>
<td>10 minutes</td>
</tr>
<tr>
<td>3</td>
<td><strong>E1: Problem 1 (WEs)</strong></td>
<td><strong>E2: Problem 1 (no WEs)</strong></td>
<td>45 minutes</td>
</tr>
<tr>
<td>4</td>
<td><strong>Post-test1</strong> + Cognitive load survey</td>
<td></td>
<td>25 minutes</td>
</tr>
<tr>
<td></td>
<td><strong>Second Day</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Re-introduce the Peer Code Helper</td>
<td></td>
<td>5 minutes</td>
</tr>
<tr>
<td>6</td>
<td><strong>E1: Problem 2 (no WEs)</strong></td>
<td><strong>E2: Problem 2 (WEs)</strong></td>
<td>45 minutes</td>
</tr>
<tr>
<td>7</td>
<td><strong>Post-test2</strong> + Cognitive load survey</td>
<td></td>
<td>25 minutes</td>
</tr>
<tr>
<td>8</td>
<td>Problem 3 (Brick Wall, no WEs)</td>
<td></td>
<td>45 minutes</td>
</tr>
<tr>
<td>9</td>
<td>Demographics (post-survey) + Cognitive load survey</td>
<td></td>
<td>15 minutes</td>
</tr>
</tbody>
</table>
Pre-test and Post-tests Examples

a, b, and temporary are variables. What does this program do?

1. Makes a and b equal to each other
2. Rearranges the variables a, b, and temporary
3. This script does not do anything
4. Swaps the values of a and b

To ensure the value of x is 15 and y is 10 after running this script, which block is missing in the blocks below?

Adapted from the Commutative Assessments (Weintrop & Wilensky, 2015)
# Study Outline

<table>
<thead>
<tr>
<th>Step</th>
<th>Group E1</th>
<th>Group E2</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Snap! Introduction (taught by camp instructor)</td>
<td></td>
<td>90 minutes</td>
</tr>
<tr>
<td>1</td>
<td>Experience pre-survey + Knowledge pre-test</td>
<td></td>
<td>35 minutes</td>
</tr>
<tr>
<td>2</td>
<td>Introduce the Peer Code Helper</td>
<td></td>
<td>10 minutes</td>
</tr>
<tr>
<td>3</td>
<td><strong>E1: Problem 1 (WEs)</strong></td>
<td><strong>E2: Problem 1 (no WEs)</strong></td>
<td>45 minutes</td>
</tr>
<tr>
<td>4</td>
<td>Post-test1 + <strong>Cognitive load survey</strong></td>
<td></td>
<td>25 minutes</td>
</tr>
<tr>
<td></td>
<td><strong>Second Day</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Re-introduce the Peer Code Helper</td>
<td></td>
<td>5 minutes</td>
</tr>
<tr>
<td>6</td>
<td><strong>E1: Problem 2 (no WEs)</strong></td>
<td><strong>E2: Problem 2 (WEs)</strong></td>
<td>45 minutes</td>
</tr>
<tr>
<td>7</td>
<td>Post-test2 + <strong>Cognitive load survey</strong></td>
<td></td>
<td>25 minutes</td>
</tr>
<tr>
<td>8</td>
<td>Problem 3 (Brick Wall, no WEs)</td>
<td></td>
<td>45 minutes</td>
</tr>
<tr>
<td>9</td>
<td>Demographics (post-survey) + <strong>Cognitive load survey</strong></td>
<td></td>
<td>15 minutes</td>
</tr>
</tbody>
</table>
Cognitive Load Survey (CS CLCS)

Intrinsic Load
1. The topics covered in the activity were very complex.
2. The activity covered program code that I thought was very complex.
3. The activity covered concepts and definitions that I thought were very complex.

Extraneous Load
4. The instructions and/or explanations during the activity were very unclear.
5. The instructions and/or explanations were very unhelpful for my learning.
6. The instructions and/or explanations were full of unclear language.

Germane Load
7. The activity really enhanced my understanding of the topic(s) covered.
8. The activity really enhanced my knowledge and understanding of computing/programming.
9. The activity really enhanced my understanding of the program code covered.
10. The activity really enhanced my understanding of the concepts and definitions.

(Morrison et al., 2014)
## Study Outline

<table>
<thead>
<tr>
<th>Step</th>
<th>Group E1</th>
<th>Group E2</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Snap! Introduction (taught by camp instructor)</td>
<td></td>
<td>90 minutes</td>
</tr>
<tr>
<td>1</td>
<td>Experience pre-survey + Knowledge pre-test</td>
<td></td>
<td>35 minutes</td>
</tr>
<tr>
<td>2</td>
<td>Introduce the Peer Code Helper</td>
<td></td>
<td>10 minutes</td>
</tr>
<tr>
<td>3</td>
<td>E1: Problem 1 (WEs)</td>
<td>E2: Problem 1 (no WEs)</td>
<td>45 minutes</td>
</tr>
<tr>
<td>4</td>
<td>Post-test1 + Cognitive load survey</td>
<td></td>
<td>25 minutes</td>
</tr>
<tr>
<td></td>
<td><strong>Second Day</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Re-introduce the Peer Code Helper</td>
<td></td>
<td>5 minutes</td>
</tr>
<tr>
<td>6</td>
<td>E1: Problem 2 (no WEs)</td>
<td>E2: Problem 2 (WEs)</td>
<td>45 minutes</td>
</tr>
<tr>
<td>7</td>
<td>Post-test2 + Cognitive load survey</td>
<td></td>
<td>25 minutes</td>
</tr>
<tr>
<td>8</td>
<td><strong>Problem 3 (Brick Wall, no WEs)</strong></td>
<td></td>
<td>45 minutes</td>
</tr>
<tr>
<td>9</td>
<td>Demographics (post-survey) + Cognitive load survey</td>
<td></td>
<td>15 minutes</td>
</tr>
</tbody>
</table>
RQ1: Student Learning

How does having access to WEs during a programming problem impact students’ learning during the problem?
Pre- and Post-tests Results

Table: Mean (with SD) pre-test, post-test1, and post-test2 scores

<table>
<thead>
<tr>
<th>Group</th>
<th>Pre-test</th>
<th>Post-test1</th>
<th>Post-test2</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1 (N=8)</td>
<td>8.13 (4.22)</td>
<td>8.38 (3.62)</td>
<td>9.13 (3.83)</td>
</tr>
<tr>
<td>E2 (N=8)</td>
<td>7.13 (4.02)</td>
<td>7.50 (3.21)</td>
<td>8.50 (4.21)</td>
</tr>
</tbody>
</table>
Pre- and Post-tests Results

Table: Mean (with SD) pre-test, post-test1, and post-test2 scores

<table>
<thead>
<tr>
<th>Group E1 (N=8)</th>
<th>Pre-test</th>
<th>Post-test1</th>
<th>Post-test2</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.13 (4.22)</td>
<td>8.38 (3.62)</td>
<td>9.13 (3.83)</td>
<td></td>
</tr>
<tr>
<td>Group E2 (N=8)</td>
<td>7.13 (4.02)</td>
<td>7.50 (3.21)</td>
<td>8.50 (4.21)</td>
</tr>
</tbody>
</table>

No significant difference on pre-test scores between groups: $t(13.96) = -0.4, p = .64, d = .24$
Pre- and Post-tests Results

Table: Mean (with SD) pre-test, post-test1, and post-test2 scores

<table>
<thead>
<tr>
<th></th>
<th>Pre-test</th>
<th>Post-test1</th>
<th>Post-test2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group E1 (N=8)</td>
<td>8.13 (4.22)</td>
<td>8.38 (3.62)</td>
<td>9.13 (3.83)</td>
</tr>
<tr>
<td>Group E2 (N=8)</td>
<td>7.13 (4.02)</td>
<td>7.50 (3.21)</td>
<td>8.50 (4.21)</td>
</tr>
</tbody>
</table>

Main effect of test: \( F(2,28) = 5.26, \ p < .05, \ \text{partial } \eta^2 = .27 \)
Pre-test to post-test2: \( t(15) = 3.05, \ p < .01, \ d = .30 \)
Post-test1 to post-test2: \( t(15) = 3.05, \ p < .01, \ d = .30 \)
Pre-test to post-test1: \( t(15) = -0.86, \ p = .40, \ d = .08 \)
No main effect of group: \( F(1,14) = 0.20, \ p = .66, \ \text{partial } \eta^2 = .014 \)
No significant interaction between group and test: \( F(2,28) = 0.13, \ p = .88, \ \text{partial } \eta^2 = .009 \)
RQ1: Student Learning

How does having access to WE platforms during a programming problem impact students’ learning during the problem?

- Most of students' learning occurred during problem 2
  - Having time to reflect and digest the concepts learned in problem 1
- We did not find significant differences in learning between groups on the WE problems
  - Most students completed the core objectives
RQ2: Cognitive Load

How does having access to WEs during a programming problem impact students’ perceived difficulty and cognitive load with respect to the problem?
Cognitive Load Survey Results

Table: Mean (SD) factor score of cognitive load (IL - Intrinsic Load, EL - Extraneous Load, GL - Germane Load)

<table>
<thead>
<tr>
<th></th>
<th>Problem 1 - Daisy Design</th>
<th>Problem 2 - Spiral Polygon</th>
<th>Problem 3 - Brick Wall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IL (N = 8)</td>
<td>EL (N = 8)</td>
<td>IL (N = 8)</td>
</tr>
<tr>
<td>Group E1</td>
<td>4.3 (2.4)</td>
<td>3.5 (3.7)</td>
<td>6.6 (3.2)</td>
</tr>
<tr>
<td></td>
<td>6.4 (2.4)</td>
<td>3.3 (2.8)</td>
<td>6.7 (2.4)</td>
</tr>
<tr>
<td></td>
<td>5.3 (3.3)</td>
<td>3.4 (2.5)</td>
<td>7.9 (2.6)</td>
</tr>
<tr>
<td>Group E2</td>
<td>4.9 (2.6)</td>
<td>3.4 (2.6)</td>
<td>8.6 (1.6)</td>
</tr>
<tr>
<td></td>
<td>3.8 (1.9)</td>
<td>2.7 (1.7)</td>
<td>8.0 (2.1)</td>
</tr>
<tr>
<td></td>
<td>6.3 (2.8)</td>
<td>4.9 (3.5)</td>
<td>7.6 (2.3)</td>
</tr>
</tbody>
</table>

No main effect of group: \((F(1,14) = 0.10, p = .76, \text{ partial } \eta^2 = .007)\)
No main effect of problem: \((F(2, 28) = 1.78, p = .19, \text{ partial } \eta^2 = .011)\)

Significant interaction between group and problem: \((F(2,28) = 4.65, p = .05, \text{ partial } \eta^2 = .25)\)
Cognitive Load Survey Results

Table: Mean (SD) factor score of cognitive load (IL - Intrinsic Load, EL - Extraneous Load, GL - Germane Load)

<table>
<thead>
<tr>
<th></th>
<th>Problem 1 - Daisy Design</th>
<th>Problem 2 - Spiral Polygon</th>
<th>Problem 3 - Brick Wall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IL</td>
<td>EL</td>
<td>GL</td>
</tr>
<tr>
<td>Group E1 (N = 8)</td>
<td>4.3 (2.4)</td>
<td>3.5 (3.7)</td>
<td>6.6 (3.2)</td>
</tr>
<tr>
<td>Group E2 (N = 8)</td>
<td>4.9 (2.6)</td>
<td>3.4 (2.6)</td>
<td>8.6 (1.6)</td>
</tr>
</tbody>
</table>

E1P1 vs. E2P1: \( t(13.91) = 0.40, p = .69, d = 0.20 \)
E1P2 vs. E2P2: \( t(13.19) = -2.33, p < .05, d = -1.16 \)

Why?
# Cognitive Load Survey Results

## Table: Mean (SD) factor score of cognitive load (IL - Intrinsic Load, EL - Extraneous Load, GL - Germane Load)

<table>
<thead>
<tr>
<th></th>
<th>Problem 1 - Daisy Design</th>
<th>Problem 2 - Spiral Polygon</th>
<th>Problem 3 - Brick Wall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IL</td>
<td>EL</td>
<td>GL</td>
</tr>
<tr>
<td>Group E1 (N = 8)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.3 (2.4)</td>
<td>3.5 (3.7)</td>
<td>6.6 (3.2)</td>
</tr>
<tr>
<td>Group E2 (N = 8)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.9 (2.6)</td>
<td>3.4 (2.6)</td>
<td>8.6 (1.6)</td>
</tr>
</tbody>
</table>

**Possible explanations:**
- WEs reduce intrinsic load
- WEs represent an inherently different learning task than problem solving
- The self-reported instrument may not be accurate
### Cognitive Load Survey Results

Table: Mean (SD) factor score of cognitive load (IL - Intrinsic Load, EL - Extraneous Load, GL - Germane Load)

<table>
<thead>
<tr>
<th></th>
<th>Problem 1 - Daisy Design</th>
<th>Problem 2 - Spiral Polygon</th>
<th>Problem 3 - Brick Wall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IL</td>
<td>EL</td>
<td>GL</td>
</tr>
<tr>
<td>Group E1 (N = 8)</td>
<td>4.3 (2.4)</td>
<td>3.5 (3.7)</td>
<td>6.6 (3.2)</td>
</tr>
<tr>
<td>Group E2 (N = 8)</td>
<td>4.9 (2.6)</td>
<td>3.4 (2.6)</td>
<td>8.6 (1.6)</td>
</tr>
</tbody>
</table>

E1P1 vs. E1P2: $t(7) = -3.51, p < .01, d = 0.83$
E2P2 vs. E2P3: $t(7) = -4.52, p < .01, d = 1.04$
## Cognitive Load Survey Results

Table: Mean (SD) factor score of cognitive load (IL - Intrinsic Load, EL - Extraneous Load, GL - Germane Load)

<table>
<thead>
<tr>
<th></th>
<th>Problem 1 - Daisy Design</th>
<th>Problem 2 - Spiral Polygon</th>
<th>Problem 3 - Brick Wall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IL</td>
<td>EL</td>
<td>GL</td>
</tr>
<tr>
<td>Group E1 (N = 8)</td>
<td>4.3 (2.4)</td>
<td>3.5 (3.7)</td>
<td>6.6 (3.2)</td>
</tr>
<tr>
<td>Group E2 (N = 8)</td>
<td>4.9 (2.6)</td>
<td>3.4 (2.6)</td>
<td>8.6 (1.6)</td>
</tr>
</tbody>
</table>

WEs may increase students' perceived difficulty of problem solving immediate following WEs
RQ2: Cognitive Load

How does having access to WEs during a programming problem impact students’ perceived difficulty and cognitive load with respect to the problem?

- We found significant differences between the groups' intrinsic cognitive load for problem 2 but not for problem 1
- We also found both groups experienced higher intrinsic load on problems without WEs that followed problems with WEs
RQ3: Programming Efficiency

How does having access to WEs during a programming problem impact students’ programming efficiency?
Objectives Completed Over Time

The average number of objectives completed by each group over time, with shading indicating ±1 standard error.
Objectives Completed Over Time

The average number of objectives completed by each group over time, with shading indicating ±1 standard error.

Rui Zhi, Thomas W. Price, Samiha Marwan, Alexandra Milliken, Tiffany Barnes, Min Chi
Objectives Completed Over Time

The average number of objectives completed by each group over time, with shading indicating ±1 standard error.

Rui Zhi, Thomas W. Price, Samiha Marwan, Alexandra Milliken, Tiffany Barnes, Min Chi
If We Cut the Time ...

The average number of objectives completed by each group over time, with shading indicating ±1 standard error.
RQ3: Programming Efficiency

How does having access to WEs during a programming problem impact students’ programming efficiency?

- Our analysis suggests that WEs save students considerable time in completing programming objectives, but that students take longer to complete later objectives.
Post-survey Feedback

Would you like to have the Peer Code Helper on future programming activities?

- 35% yes
  - Appreciate the PCH
    - “see how to go from one step to the next”
- 12% no
  - had very high pre-test scores (over 75%)
    - More advanced students may not appreciate worked examples (Kalyuga et al., 2003)
- 53% uncertain
  - Prefer the challenge of working independently
    - “It’s good to have a challenge, but it’s also nice... to make it a little bit easier”
Conclusion

- Worked examples may have an effect on students' intrinsic cognitive load

- Programming worked examples may improve students' programming efficiency in the short term, but that students do require additional time to process WEs before they can construct their own code
Thank you for your time!

Questions?

Rui Zhi
rzhi@ncsu.edu

Advisor:
Dr. Tiffany Barnes
Dr. Thomas W. Price
Cognitive Load Survey (CS CLCS)

Intrinsic Load
1. The topics covered in the activity were very complex.
2. The activity covered program code that I thought was very complex.
3. The activity covered concepts and definitions that I thought were very complex.

Extraneous Load
4. The instructions and/or explanations during the activity were very unclear.
5. The instructions and/or explanations were very unhelpful for my learning.
6. The instructions and/or explanations were full of unclear language.

Germane Load
7. The activity really enhanced my understanding of the topic(s) covered.
8. The activity really enhanced my knowledge and understanding of computing/programming.
9. The activity really enhanced my understanding of the program code covered.
10. The activity really enhanced my understanding of the concepts and definitions.

(Morrison et al., 2014)