

# Final Reading Training Reflection and Data Collection

This is the final qualitative record for the reading training sessions. Note: All questions are optional. If you feel uncomfortable answering something, please leave it blank

The respondent's email (**null**) was recorded on submission of this form.

1. Email \*

---

2. What is your name?

---

3. In three to five sentences, please describe three strategies you could use to approach reading technical writing.

---

---

---

---

---

4. If you only have 10-15 minutes to scan a paper or article, how can you get the most information out of the reading in a limited time? (e.g. what sections can you look at? What speed reading strategies can you use?)

---

---

---

---

---

5. What is API Documentation? Why is it helpful? How do you use it?

---

---

---

---

---

6. Which three sessions/topics/papers did you find the most interesting?

*Check all that apply.*

- Affixes and Roots
- Vocabulary
- Workbook Warmup
- Heilmeier Catechism
- Speed Reading Papers
- Understanding Charts and Figures
- API Documentation
- Paper: Applying a gesture taxonomy to introductory computing concepts
- Paper: On the Naturalness of Software
- Paper: InFix: Automatically Repairing Novice Program Inputs
- Paper: Youth Computational Participation in the Wild: Understanding Experience and Equity in Participating and Programming in the Online Scratch Community
- Paper: Aggregate Compilation Behavior: Findings and Implications from 27,698 Users
- Paper: What makes a Great Manager of Software Engineers
- Paper: Debugging Support for Pattern-Matching Languages and Accelerators
- Paper: Synthesizing API Usage Examples

7. Which three sessions/topics/papers did you find the least interesting?

*Check all that apply.*

- Affixes and Roots
- Vocabulary
- Workbook Warmup
- Heilmeier Catechism
- Speed Reading Papers
- Understanding Charts and Figures
- API Documentation
- Paper: Applying a gesture taxonomy to introductory computing concepts
- Paper: On the Naturalness of Software
- Paper: InFix: Automatically Repairing Novice Program Inputs
- Paper: Youth Computational Participation in the Wild: Understanding Experience and Equity in Participating and Programming in the Online Scratch Community
- Paper: Aggregate Compilation Behavior: Findings and Implications from 27,698 Users
- Paper: What makes a Great Manager of Software Engineers
- Paper: Debugging Support for Pattern-Matching Languages and Accelerators
- Paper: Synthesizing API Usage Examples

8. Which three sessions/topics/papers did you find the most helpful?

*Check all that apply.*

- Affixes and Roots
- Vocabulary
- Workbook Warmup
- Heilmeier Catechism
- Speed Reading Papers
- Understanding Charts and Figures
- API Documentation
- Paper: Applying a gesture taxonomy to introductory computing concepts
- Paper: On the Naturalness of Software
- Paper: InFix: Automatically Repairing Novice Program Inputs
- Paper: Youth Computational Participation in the Wild: Understanding Experience and Equity in Participating and Programming in the Online Scratch Community
- Paper: Aggregate Compilation Behavior: Findings and Implications from 27,698 Users
- Paper: What makes a Great Manager of Software Engineers
- Paper: Debugging Support for Pattern-Matching Languages and Accelerators
- Paper: Synthesizing API Usage Examples

9. Which three sessions/topics/papers did you find the least helpful?

*Check all that apply.*

- Affixes and Roots
- Vocabulary
- Workbook Warmup
- Heilmeier Catechism
- Speed Reading Papers
- Understanding Charts and Figures
- API Documentation
- Paper: Applying a gesture taxonomy to introductory computing concepts
- Paper: On the Naturalness of Software
- Paper: InFix: Automatically Repairing Novice Program Inputs
- Paper: Youth Computational Participation in the Wild: Understanding Experience and Equity in Participating and Programming in the Online Scratch Community
- Paper: Aggregate Compilation Behavior: Findings and Implications from 27,698 Users
- Paper: What makes a Great Manager of Software Engineers
- Paper: Debugging Support for Pattern-Matching Languages and Accelerators
- Paper: Synthesizing API Usage Examples

10. What grade do you anticipate receiving in EECS 183?

*Mark only one oval.*

- A+
- A
- A-
- B+
- B
- B-
- C+
- C
- C-
- D
- F
- Pass
- No Record - COVID
- Withdrew
- Prefer not to say

11. Do you plan on continuing with computer science?

*Mark only one oval.*

- Yes
- No

12. If yes, how do you plan to do so?

*Check all that apply.*

- I plan on majoring in computer science
- I plan on minoring in computer science
- I plan on majoring in the School of Information
- I plan on majoring or minoring in data science
- I plan on attending graduate school for computer science
- I plan on continuing to learn computer science on my own

13. What are your major(s) / minors(s)? Or what major(s) and minor(s) are you planning on declaring?

---

14. Did taking EECS 183 increase or decrease your desire to major or minor in computer science?

*Mark only one oval.*

- Increase
- Decrease
- No Change

15. Did participating in the Transfer Training Study increase or decrease your desire to major or minor in computer science?

*Mark only one oval.*

- Increase
- Decrease
- No Change

16. Do you personally feel that the reading training sessions helped you with EECS 183 (either with the class itself or with learning how to code)?

*Mark only one oval.*

- Yes
- No
- Other: \_\_\_\_\_

17. If yes, please describe how you found the training helpful for EECS 183

---

---

---

---

---

18. If no, what would you have changed to make the course more helpful?

---

---

---

---

---

19. Would you be interested in being contacted for participating in any follow-up studies (either looking at coding retention or to do additional fNIRS scans in the fall)?

*Mark only one oval.*

- Yes
- No



20. Do you have any other additional comments for the research team?

---

---

---

---

---

---

This content is neither created nor endorsed by Google.

Google Forms