

Homework 04 - Interviews and Content Logs

Andrew Carter and Beryl Egerter

February 20, 2013

1 Interview: ACBE1

1.1 Post Interview Reflection (Joint)

- Hard to tell when participant is asking questions we should answer.
- Did first two questions prime participant too much to completely understand the code instead of trying to fix specific problem?
- Always started from a top of file
- When functions were small, would skip around if a function was called, but not as much skipping around when looking at larger functions.
- Got lost in code, think forgot what initial question was - we want to provide printed question to participant in future interviews in order to try and make sure they remember what they are doing
- Often when confused, didn't have a specific question that we could answer without leading too much
- Ruby question actually worked well because we knew more when we should step in and explain a ruby question because participant hadn't seen ruby before
- Occasionally missed pieces of the code (eg. a plus sign between two function calls)
- Got bogged down in functions that didn't matter to question - wanted to understand code fully which we believe contributed to forgetting initial question in 2.2

1.2 Interview Summary

Interview length: 49:20m

Technical difficulties:

- Table is reflective

- Paper was missing a new page mark, so two questions had strange paper size
- Double equals doesn't have a break in the middle because of the font

We went through Questions 1.1, 1.2, 2.1, and 2.2.

1.3 Most Interesting Pieces

- Time: 10:00 - 12:00 (Video 1)
Summary: SUBJECT partially calculates result of a function call for func188 in 1.2. However SUBJECT forgets to wrap in a list and add the first half, SUBJECT then goes back, points out it returns some of list (which SUBJECT still forgets) then realizes second value is also in a list and moves on.
Relations to Topic: SUBJECT is trying to work through evaluation of code they haven't seen before and getting confused.
Possible Arguments: While debugging, it is easy for programmer to introduce new bugs.
- Time: 2:12 - 3:56, 4:41 - 6:15ish (Video 1)
Summary: SUBJECT points out "f" is bound to func4 in a call of func1, proceeds to forget this binding, later asks what's "f", then says "ohhh" but is still very confused. Retries later, still has trouble figuring out what "f" was bound to, even though she clearly pointed it out in the very first pass.
Relations to Topic: SUBJECT is tracing through code and getting confused about portions already figured out.
Possible Arguments: The act of trying to interpret other peoples code involves keeping many things in mind.

1.4 Content Log

Content log is provided in separate text file labeled ACBE1.

2 Interview: ACBE2

2.1 Post Interview Reflection (Joint)

- Went through first two problems very efficiently
- In 2.1, figured out problem with tomorrow and how to fix it, and what the problem with yesterday was, but not how to fix yesterday until prompted
- Much less thrown off by ruby than previous participant
- Could infer some syntax correctly (inferred some incorrectly) but was still able to work with the code without being told that her assumptions about syntax were incorrect/correct
- backtraced on the smaller functions, skipped str function but not score in 2.2
- didn't backtrace on 2.2 even though there was code that would backtrace similarly to the first two problems
- needed to understand whole class, not just specific functions within it
- first question was put off by passing functions but recovered quickly
- like previous participant, replaced function calls with results but only in comprehension
- relied on names a lot in second two questions

2.2 Interview Summary

Interview length: 36:25m

Technical difficulties:

- Camera setting were slightly different, but recording came out fine
- One function in Question 2.2 was on two pages, since it was in python, indentation was actually important

We went through Questions 1.1, 1.2, 2.1, and 2.2.

2.3 Most Interesting Pieces

- Time: 13:15 - 13:55
Summary: SUBJECT looks at tomorrow function, recognizes that there is a part of the function that we care about in relation to finding the 'bug'.
Relations to Topic: SUBJECT has found a piece of information within a file and knows that they care about it, a good step when looking at code you are unfamiliar with but need to bug fix.
Possible Arguments: A programmer does not need to understand a whole file in order to find important pieces of code.

- Time: 4:37 - 4:55, 19:00 - 19:24

Summary: In first segment, SUBJECT explains that they started with `main()` so they could backtrack since they had no context for the other functions. In second segment, SUBJECT explains that with the class based code and the longer code they couldn't just go from function to function, had to understand what everything was doing, was different than the code talked about in the first segment.

Relations to Topic: Showing different ways that a programmer first looks at unfamiliar code.

Possible Arguments: The concepts used (Object Oriented v. functional) as well as style of code (long v. small functions) can affect how the programmer perceives how they should go about analyzing the code.

2.4 Content Log

Content log is provided in separate text file labeled ACBE2.

3 Next Interviews

Our scheduled interviews:

1. Interview with EG at 10:00am on Thursday, February 21
2. Interview with DE at 9:00pm on Tuesday, February 26