

Homework 01 - Topic Selection

Andrew Carter and Beryl Egerter

January 29, 2013

1 Topic

Often, whether in school or industry, programmers are given code that they have not seen before. Familiarizing themselves with that code can take varying amounts of time. We would like to investigate how this process occurs and what factors affect it. By learning how people reason about unfamiliar code and the results of their techniques we can help programmers be more prepared and confident in these situations.

2 Interview Questions

Code samples for interview questions are located on Github. The short sample for our first task/question is included on this document, since the others are longer, we leProviding a correct answer to a question is relatively unimportant compared to the process the interviewee goes through to answer it. Following is an explanation of three tasks relating to code samples we believe could help us understand our topic.

1. The first code sample, in Python, is very short and simple in constructs, though the order of evaluation is complex. Seeing how an interviewee familiarizes his or herself with the code particularly in the order in which the functions are analyzed is of interest.

```
def func2(list, num):  
    return func1(list, num, func4)  
  
def func4(a, b):  
    return a * b  
  
def func1(list, num, f):  
    acc = 0  
    for i in list:  
        acc += f(i, num)  
    return acc
```

```
def main():
    print(func3([1,2,3,4]))

def func3(list)
    return func2(list, 4)

main()
```

2. The second code sample, in Ruby, a language the student does not know, is a class they have written in CS5, the Date class. We will give the interviewee the following task: "A coworker recently left on vacation and left two files behind. One of those files is a Date class and your boss wasn't sure if the coworker had finished including leap year support. Your boss would like you to make sure it is supported." We will mainly be looking at what order the interviewee looks at functions or files, and whether or not they identify the locations in which code needs to be added.
3. For the third code sample, in C, the interviewee is expected to know C++ and have some idea of preprocessor macros. They will be given an RPN calculator that uses macros, where one of the macros has unbalanced parentheses. We will give the interviewee the following task: "That coworker also had built an RPN calculator, but your boss accidentally made a small change to the file, but he isn't sure what he did, and now it won't compile. Can you take a look at it, and see if you can find the change?" We will mainly be looking at debugging strategies for compile errors, the code was designed to have a relatively unhelpful compile message.

3 Initial Hypothesis

We hypothesize that people will use different techniques to familiarize themselves with code. We believe that programmers struggle with this topic because different code can have different idioms, similar to learning foreign languages where simply learning the words is not enough to understand someone speaking. Even when proficient with a programming language, programmers may need to learn a new coding style to familiarize themselves with new code. A source of misunderstanding about this topic might be the degree to which people familiarize themselves with code. For example, knowing code well enough to say what it does is different than knowing the code well enough to modify its function.

4 Target Interview Population

We would like to interview Computer Science majors at Harvey Mudd who have taken at least CS70, and are at the same point within the Computer Science curriculum to get a homogeneous population sample due to the small sample size we will have for these interviews. We would also like the interviewees to not know Ruby.