

Teacher Interview Protocol

0. General information

Male/Female

Where do you teach? (Your institution)

In which field are your students specializing?

What subject do you teach?

How long have you been teaching?

What is your academic background?

1. What programming language do you use to introduce the basics of programming?
2. What are the most challenging key programming concepts in your teaching perspective?
3. What is the major learning obstacle that students face before being introduced to object-oriented programming?
[Answers to question 2 may include OO concepts, but here the intended focus is on imperative and procedural concepts.]
4. In which order do you teach the basic programming-related concepts?
5. How much time do you plan for each basic concept?
6. Can you show some of your favorite examples to make students learn how to apply the iteration constructs?
7. Are the tasks assigned to students simple variations of those dealt with in class? Or do they address unfamiliar situations as well?
8. What are the extra-computing prerequisites necessary to understand the basic programming concepts as well as the examples you show?
9. In your experience, to what extent can students master the termination condition of a loop?
10. In your teaching, do you cover the mappings between different iteration constructs (*for*, *while*, *do-while*, *repeat-until*)?
11. How do you usually assess an incorrect termination condition? And oversights about the first or last iteration?
12. How do you assess a working solution if it is inefficient, or convoluted, or somehow at odds with what you expected?
13. While trying to achieve the assigned tasks, do you expect your students to apply the models introduced in class? Or do you also appreciate “creative” solutions?
14. Which features of the iteration constructs are usually understood by (most) students, and which are more difficult to them?
15. What are your more frequent suggestions to students for improving their programming performance?
16. Are the different solutions proposed by students compared in class? How?
17. What are your strategies to motivate students?
18. How do you attempt to manage different learning styles?
19. How do you deal with students’ criticisms in order to address their possible needs?
20. Any other issues you deem important to consider about the teaching/learning of programming?