Some questions that could be asked during oral exam:

- How adaptable are the techniques presented here to JavaScript, Python, and other high-level scripting languages, which are prevalent today? In the case of JavaScript, how would you handle events? Will techniques from Chapter 4 work here?
- How does your technique in Chapter 4 compare to the-state-of-the-art concurrent program testing that focus on schedule-exploration and data-race detection (e.g. Chess, Goldilocks, Event-racer)? They're definitely not greybox. If your technique is substantially better, some experimental comparisons should be made
- What's the complexity of your algorithm in Theorem 6.46?
- You mentioned in conclusion that want to combine testing and verification to obtain a better technique for software security. How do you plan to go about doing this?