Advanced Programming Languages
Homework Assignment 0F

EECS 590

Logistics. You must work alone. Your name and Michigan email address must appear on the first page of your PDF submission but *may not appear anywhere else*. This is to protect your identity during peer review. The first page of your submission is *not* shared during peer view but all subsequent pages are.

Submission. Questions with F labels are answered via formal prose writeups and are submitted via Gradescope. These are typically manually graded. Questions with C labels require code submissions to autograder.io and are typically automatically graded. We always use your highest-scoring autograder.io submission for your final grade (so if you mistakenly submit a lower-scoring submission later, that does not hurt you). Note that formal writeup questions may require you to write or run programs (as in 0F-3 below) and the code you submit is often based on the theory from class. Homework 0 does not have an autograder.io submission but does require you to run a program.

Exercise 0F-1. Bookkeeping [5 points]. These answers should appear on the first page of your submission and are kept private.

1. Why are you taking the class and what do you hope to learn? (Any non-trivial answer is full credit. If you just need the credits and the course looked easy, that's fine. The class may be adjusted slightly toward what the majority of students want, so there is no benefit in saying you want intense formalisms if you're looking for an easy semester and no point in saying you want a gentle skim if you're considering a future career in programming systems.)

2. Pose a question that you would like to see addressed in class or in office hours. (Any non-trivial answer is full credit. Students often ask questions relating to career goals or concepts they would like to understand in the course, but anything is fair game.)

3. Tell me something about yourself that I do not already know. (Any non-trivial answer is full credit. Student interests are sometimes incorporated into in-class trivia.)

4. How would you write an extended regular expression to identify words containing three adjacent doubled letters (e.g., "bookkeeping", "sweettooth", etc.)? Support or

refute the claim that extended regular expressions are equivalent to deterministic finite automata.

Exercise 0F-2. Set Theory [5 points]. This answer should appear after the first page of your submission and may be shared during class peer review.

This exercise is meant to help you refresh your knowledge of set theory and functions. Let X and Y be sets. Let $\mathcal{P}(X)$ denote the powerset of X (the set of all subsets of X). There is a 1-1 correspondence (i.e., a bijection) between the sets A and B, where $A = X \to \mathcal{P}(Y)$ and $B = \mathcal{P}(X \times Y)$. Note that A is a set of functions and B is a (or can be viewed as a) set of relations. This correspondence will allow us to use functional notation for certain sets in class. This is Exercise 1.4 from page 8 of the Winskel textbook.

Demonstrate the correspondence between A and B by presenting an appropriate function and proving that it is a bijection. For example, you might construct a function $f : B \to A$ and prove that f is an injection and a surjection.

Exercise 0F-3. Model Checking [10 points]. This answer should appear after the first page of your submission and may be shared during class peer review.

Download the CPAChecker software model-checking tool using the instructions on the homework webpage. Read through enough of the manual to run the tool on the `tcas.i` testcase provided on the homework webpage. Check the three properties given. For each command, copy or screenshot the last ten non-empty lines of output from CPAChecker and include them as part of your answer to this question.

In this graduate-level class, it is your responsibility to get CPAChecker up and running properly. This may require you to set up a virtual machine. This level of systems programming experience is a prerequisite for the class and is intentionally part of an early assignment (i.e., before the drop deadline) to help students determine if they have the right incoming preparation.

Hint: if your output when checking `Property1a` does not indicate something like "Verification result: FALSE. Property violation (error label in line 1963) found by chosen configuration." then you may not have set things up correctly.

In at most three paragraphs, summarize your experience with the CPAChecker tool.

- What is going on when you run CPAChecker using the commands listed? What does `Property1a` mean? Is `tcas.i` a reasonable test suite? What has been proved? (This is the heart of the question. You may have to read ugly code, understand a legacy tool, and apply concepts from class. It is expected that answering this well will require time.)

- Did you find CPAChecker to be a usable tool? How easy is it to provide the inputs to CPAChecker? What information is present in the graphical (HTML) output?

For full credit, do not restate the lecture on counter-example guided abstraction refinement; instead, discuss your thoughts and experience using this tool (including its input

requirements, output guarantees, and context). Focus on threats to validity (e.g., imagine that you were writing a paper and using this as an experiment) over usability.

Both your ideas and also the clarity with which they are expressed (i.e., your English prose) matter. You can use a tool like ChatGPT to refine your prose, but be wary of false claims. A reader should be able to identify your main points, the arguments you are making, and your conclusion.

Submission. Turn in your assignment as a single PDF document via the `gradescope` website. Your name and Michigan email address must appear on the first page of your PDF submission but may not appear anywhere else.