Project - CSE 105 Spring 2022

CSE105Sp22

Part 2 due 5/19/22 at 5pm (no penalty late submission until 8am next day)

The project component of this class will be an opportunity for you to extend your work on assignments and explore applications of your choosing.

Why? To go deeper and explore the material from Theory of Computation and how it relates to other aspects of CS and beyond.

How? During emergency remote instruction last academic year, we discovered that video assessment and some open-ended personalized projects help ensure fairness and can be less stressful for students than in-person midterm exams. Asynchronous project submission also gives flexibility and allows more physical distancing.

Your videos: We will delete all the videos we receive from you after assigning final grades for the course, and they will be stored in a university-controlled Google Drive directory only accessible to the course staff during the quarter. Please send an email to the instructor (minnes@eng.ucsd.edu) if you have concerns about the video / screencast components of this project or cannot complete projects in this style for some reason.

You may produce screencasts with any software you choose. One option is to record yourself with Zoom; a tutorial on how to use Zoom to record a screencast (courtesy of Prof. Joe Politz) is here: Tutorial URL The video that was produced from that recording session in Zoom is here: Video produced in tutorial .

What resources can you use? This project must be completed individually, without any help from other people, including the course staff (other than logistics support if you get stuck with screencast). You can use any of this quarter's CSE 105 offering (notes, readings, class videos, supplementary videos, homework feedback). You may additionally search online to respond to project parts that explicitly ask you to do so, and you must cite all resources (online or offline) that you consult as part of this search. Link directly to the resource and include the name of the author / video creator and the reason you consulted this reference. The work you submit for the project needs to be your own.

The written portion of the project is expected to be clearly legible, and should preferably be typed.

1 Tasks for Project Part 2

1.1 Task 1: Explain a review quiz question (Written)

1. Select one question from one of the review quizzes from 4/15/22 (Friday of Week 3) to 4/29/22 (Friday of Week 5) to revisit. Include the problem description, why you picked this question (e.g. what is interesting about it, what is hard about it, or why you wanted to take a second look at it), and your solution. Question selection: you can pick any one question listed in the Gradescope review quizzes, and you must address all of its parts.

2. For each part of your chosen question: prepare a complete solution (you can use the homework solutions we post for guidance about the style). Your submission will be evaluated not only on the correctness of your answers, but on your ability to present your ideas clearly and logically. You should explain how you arrived at your conclusions, using mathematically sound reasoning. Your goal should be to convince the reader that your results and methods are sound. Imagine you are preparing these solutions for someone else taking CSE 105 who missed that week and is "catching up"

3. Include at least 2 potential mistakes that a student may have made while attempting to solve the quiz problem that you selected. Explain why the reasoning behind these mistakes is flawed so that a student reading this may learn from these mistakes. It's a good idea to include mistakes that you made when you first tried to solve this problem!

Style guidelines: your written submission for Task 1 should clearly label the three parts: *Question Selection*, *Solution*, and *Potential Mistakes*.

1.2 Task 2: Closure of the Collection of Regular Languages

For this task, we fix $\Sigma = \{0, 1\}$. Recall that the composition of two functions f and g is denoted $f \circ g$, which can also be written as $f(g(x))$, and is the result of first applying the function g to the input x (producing $g(x)$), and then applying f to $g(x)$. Below are some functions with domain and codomain $\mathcal{P}(\Sigma^*)$; that is, they each take in a language over Σ and output a language over Σ.

$$TZ(L) = \{w0^k \mid w \in L, k \geq 0\}$$
$$R(L) = \{w \mid w^R \in L\} \text{(where } w^R \text{ is reversing } w, \text{ e.g. } (100)^R = 001)$$
$$E(L) = \{w \mid w \in L, \text{ the length of } w \text{ is even}\}$$
$$T(L) = \{w \mid w \in L, \text{ the length of } w \text{ is a multiple of 3}\}$$
$$EQ(L) = \{w^k 1^k \mid k \geq 0, w \in L\} \text{(where } w^k \text{ is concatenating } w \text{ with itself } k \text{ times, e.g. } (100)^2 = 100100)$$
$$LT(L) = \{w^k 1^j \mid 0 \leq k < j, w \in L\}$$
$$EQ2(L) = \{w^k x^k \mid k \geq 0, w \in L, x \in L\}$$

For example $(TZ \circ R)(L) = TZ(R(L)) = \{w^R 0^k \mid w \in L, k \geq 0\}$.

1. Choose two functions from the above list so that their composition, h, is such that the collection of regular languages over Σ is closed under h.

   (a) Provide a clear and complete definition of your function h. A complete solution will clearly specify the two functions you chose to compose, the order in which h applies them, and a general description of what the function h does using set builder descriptions and/or English prose. **Note:** You do not need to apply h to a language in this step, you only need to define h.

   (b) Prove that the collection of regular languages over Σ is closed under h by writing out the following argument in detail: Consider an arbitrary regular language L over the alphabet $\Sigma = \{0, 1\}$. Since it is regular, it is recognized by a DFA and let $M = (Q, \Sigma, \delta, q_0, F)$ be such a DFA over Σ with $L(M) = L$. Give the formal construction of an NFA N with $L(N) = h(L)$ for your function h. Briefly justify this construction by tracing the computations of N and/or referencing constructions we discussed in class and in the book. In particular, explain the role of each parameter in the definition of N in the construction.

2. Choose two functions from the above list so that their composition, h', is such that the collection of regular languages over Σ is **not** closed under h'. *Note the functions you choose for this part may or may not overlap with those from the previous part; it's up to you to decide.*

(a) Provide a clear and complete definition of your function h'. A complete solution will clearly specify the two functions you chose to compose, the order in which h' applies them, and a general description of what the function h' does using set builder descriptions and/or English prose.

(b) Give a witness language L that can be used to prove that the class of regular languages over Σ is not closed under h'. To do so: (1) clearly define a language L over Σ, (2) prove that L is regular, and (3) prove that $h'(L)$ is not regular. You may use results proved in class and / or the relevant sections in the textbook as part of your proofs if you would like, but you must label these results and provide references to the day we discussed them and/or the page number in the book.

1.3   Task 3: Implementation examples and Video

With the introduction of PDAs, our models of computation begin to approach the power that modern day computers have. Choose a specific non-regular but context-free language mentioned in some question in the review quizzes between 4/15/22 (Friday of Week 3) to 4/29/22 (Friday of Week 5); you will write a program in Java, Python, JavaScript, or C++ which is able to test membership of strings in that language. The program you write should function like a PDA, using a constant amount of memory plus access to a stack, and should only make a single pass through the string.

Presenting your reasoning and demonstrating it via screenshare are important skills that also show us a lot of your learning. Getting practice with this style of presentation is a good thing for you to learn in general and a rich way for us to assess your skills. Create a 3-5 minute screencast video with the following components:

- Start with your face and your student ID for a few seconds at the beginning, and introduce yourself audibly while on screen. You don't have to be on camera for the rest of the video, though it's fine if you are. We are looking for a brief confirmation that it's you creating the video and doing the work you submitted.

- State which language you chose from the review quiz, and show the state diagram for a PDA which recognizes the language, briefly justifying why it works.

- State which programming language you chose to use and show on the screen all the code your wrote to implement the PDA in your chosen programming language. Discuss how the behavior of your program is related to the state diagram of the PDA, and discuss the implementation choices you made when creating this program.

- Demonstrate 4 test cases (2 strings in the language recognized by your PDA, 2 strings not in this language), clearly defining each one, explaining the expected behavior of the PDA, and showing the output / feedback your program gives to indicate whether the expected behavior matches the actual behavior.

You will submit this video along with a written version of Tasks 1 and 2 to Gradescope.

Extra exploration (not for credit): What would it take to implement context-free grammars in code? Could you use any of your work from implementaing PDAs?

2 Grading Criteria and checklists

Task 1

Submission covers a complete review quiz question from the relevant weeks (all parts of the question must be addressed for multi-part questions).

Submission clearly labels review questions, including which day it's from and the problem description.

Submission includes why the student picked the problem/ what they found interesting.

Solution is written (or typed) out in detail step-by-step, with clear and correct logic and justification.

Submission includes 2 potential mistakes that a student might make while solving this question and explains why they are wrong.

Task 2

Question 1:

The function h is described clearly and completely, using appropriate notation and terminology.

The formal construction of N is clear, correct, and complete, and is justified appropriately and correctly.

Question 2:

The function h' is described clearly and completely, using appropriate notation and terminology.

The language L is specified clearly and completely and is a viable witness for the proof.

The proof that L is regular is clear, correct, and complete.

The proof that $h'(L)$ is not regular is clear, correct, and complete.

Task 3

Logistics Items

- Video loads correctly

- Video is between 3 and 5 minutes

- Video shows the student's face and ID, and they introduce themself audibly while on screen

The video clearly states which language was chosen for study, and references a specific review quiz with this language.

The video shows the state diagram of a PDA which recognizes the chosen language.

The video clearly describes which programming language was chosen for the implementaiton and gives the reasons why.

The video discusses the connections between the state diagram of the PDA and its implementation in the code.

The video clearly demonstrates all test cases, including both expected and actual output. The video should include screencasts of running the code live to demonstrate these test cases.