Homework 7: Structural Induction, Regexes, CFGs

Due date: Wednesday March 2 at 10 PM
If you work with others (and you should!), remember to follow the collaboration policy outlined in the syllabus.

In general, you are graded on both the clarity and accuracy of your work. Your solution should be clear enough that someone in the class who had not seen the problem before would understand it.

We sometimes describe approximately how long our explanations are. These are intended to help you understand approximately how much detail we are expecting. You are allowed to have longer explanations, but explanations significantly longer than necessary may receive deductions.

To help with formatting of English proofs, we've published a style guide on the website containing some tips.

Finally, be sure to read the grading guidelines for more information on what we're looking for.

1. Manhattan Walk [20 points]

Let S be a subset of $\mathbb{Z} \times \mathbb{Z}$ defined recursively as:

Basis Step: $(0, 0) \in S$
Recursive Step: if $(a, b) \in S$ then $(a, b+1) \in S$, $(a+2, b+1) \in S$

Prove that $\forall (a, b) \in S, a \leq 2b$.

Hint Remember that with structural induction you must show $P(s)$ for every element s that is added by the recursive rule – you will need to show $P()$ holds for two different elements in your inductive step.

2. What doesn't kill you makes you stronger [25 points]

Consider the following recursively defined functions:

$$f(n) = \begin{cases} 2f(n-1) & \text{if } n \in \mathbb{N}, n > 1 \\ 2 & \text{if } n = 1 \end{cases}$$

$$g(n) = \begin{cases} 2g(n-1) + 1 & \text{if } n \in \mathbb{N}, n > 1 \\ 1 & \text{if } n = 1 \end{cases}$$

Let $P(n)$ be "$g(n) < f(n)$"

(a) Imagine you wanted to prove $P(n)$ for all $n \in \mathbb{N}, n \geq 1$ by induction. Try for about 5-10 minutes to write an inductive proof and see where you get stuck. For this part, just write "I spent about X minutes trying" (as long as you report an X at least 5, you will get this point) [1 points]

(b) Give an example of numbers a, b such that $a < b$ but $2a + 1 \not< 2b$. Explain why this means it's unlikely that you would complete an inductive step to prove $P()$ here. (Hint: we just need real numbers, not necessarily integers) [2 points]

(c) Now instead, define the predicate $Q(n)$ to be "$g(n) \leq f(n) - 1$". Explain why showing $Q(n)$ for all $n \geq 1$ will also show $P(n)$ for all $n \geq 1$ (1 sentence) [2 points]

(d) Prove $Q(n)$ holds for all integers n, with $n \geq 1$ by induction. [20 points]

(e) This proof technique is called "strengthened induction" (not to be confused with strong vs. weak induction). We wanted to show $P()$, but $P()$ wasn't suitable for induction, the IH was not enough information to easily prove our target in the IS. We defined a strengthened claim, $Q()$. (we call $Q()$ "stronger" because once you know $Q()$ is true you also know $P()$ must be true). Choosing what to add is a tradeoff – the more information in $Q()$ that you add, the more information you have in your inductive hypothesis to assume. But also the more you have to show in your inductive step! You do not have to write anything for this part [0 points]

3. Recursion – See: Recursion [18 points]

For each of the following languages, give a recursive definition of the language.

Your basis step must explicitly enumerate a finite number of initial elements.

We may deduct for constructions that are needlessly complicated (e.g. more base cases than necessary, or significantly more recursive steps than necessary).

Briefly (1-2 sentences) justify that your description defines the same language. Do not give us a full proof; you do not have to justify why your description is the shortest possible.

(a) Binary strings that start with 0 and have odd length (i.e. an odd number of characters).

(b) Binary strings x such that $\texttt{len}(x) \equiv 1 \pmod 3$ where $\texttt{len}(x)$ is the number of characters in x.

(c) Binary strings with an odd number of 0s.

4. Constructing Regular Expressions (Online) [20 points]

For each of the following languages, construct a regular expression that matches exactly the given set of strings.

You will submit (and check!) your answers online at https://grin.cs.washington.edu/. Think carefully before entering a submission; you only have 10 guesses. Because these are auto-graded, we will not award partial credit.

(a) Binary strings where every occurance of a 1 is immediately followed by a 0.

(b) Binary strings where every occurence of a 0 is immediately followed by a 11

(c) Binary strings with an odd number of 1s

(d) The set of all binary strings that begin with a 1 and have length congruent to $2 \pmod 4$.

5. Context Is Everything. Except for Context-Free Grammars (Online) [15 points]

For each of the following languages, construct a context-free grammar that generates exactly the given language.

You will submit (and check!) your answers online at https://grin.cs.washington.edu/. Think carefully before entering a submission; you only have 10 guesses. Because these are auto-graded, we will not award partial credit.

(a) The set of all binary strings that contain at least one 1 and and most two 0's.

(b) $\{0^m 1^n 0^{n+2m} : m, n \geq 0\}$

(c) Binary strings with an even number of 0's

6. Feedback

Please keep track of how much time you spend on this homework and answer the following questions. This can help us calibrate future assignments and future iterations of the course, and can help you identify which areas are most challenging for you.

- How many hours did you spend working on this assignment?
- Which problem did you spend the most time on?
- Any other feedback for us?