Midterm Examination

CS 212 L2 Nature of Computation
Habib University
Oct 5, Fall 2022

35 points

Instructions:

1. This exam takes place during class time.
2. You are allowed to keep only the following items with you on your table: writing utensils, your HU ID card, your "cheat sheet", and a drinks container. Please deposit other items, e.g. bags and devices at the front of the classroom.
3. Make sure that your HU ID card is clearly visible on your table.
4. Your cheat sheet must meet the requirements shared earlier. Non-compliant cheat sheets will be confiscated.
5. Please ensure sufficient writing utensils in working condition. Do not bother others during the exam.
6. The questions in this exam rely on argumentation. Make sure to provide sound justifications that are both precise and concise.
7. If your answer to a question exceeds 10 lines, it is very likely that you are on the wrong track.
8. Please submit this question paper and your cheat sheet with your answer book.
9. This exam consists of 4 questions for 35 points, printed on 3 sides of a single sheet of paper.
10. Attempt all questions.

---

viel Spaß und viel Glück!

1. Consider car number plates generated by the following grammar.

$$S \to LLL - DDD$$
$$L \to a \mid b \mid c \mid \ldots \mid z$$
$$D \to 0 \mid 1 \mid 2 \mid \ldots \mid 9$$

For each question below, provide a Yes/No answer followed by a justification.

(a) [5 points] Is this grammar ambiguous?

Solution: No. Each string in this language has a unique parse tree.

(b) [5 points] Is this grammar in Chomsky Normal Form?

Solution: No. The first production has more than 2 variables on the right hand side.

(c) [5 points] Does the set of all such number plates constitute a regular language?

Solution: Yes. The set of all number plates that this grammar generates is finite. Therefore, it constitutes a regular language.

2. [5 points] Computers store data in binary. As such, any stored file—e.g. a program, an image, an application—can be considered a string over $\{0, 1\}$. Consider the set of all files (strings) that a given computer can store. Does this set form a regular language? If not, does it form a context-free language? Explain.

Solution: Computers have finite memory so the set of all strings that a given computer can store is finite. Hence, the set forms a regular language.

3. [10 points] We know that the class of regular languages is closed under the regular operations: union, concatenation, and star. Is the class of context-free languages also closed under these operations? Justify your answer.

Solution: Yes, the class of context-free languages is closed under the regular operations. Consider context-free languages, L_1 and L_2, whose corresponding grammars have the start symbols, S_1 and S_2.

Union The language $L = L_1 \cup L_2$ can be expressed as a CFG with the start symbol, S, such that $S \to S_1 \mid S_2$.

Concatenation The language $L = L_1 L_2$ can be expressed as a CFG with the start symbol, S, such that $S \to S_1 S_2$.

Star The language $L = L_1^*$ can be expressed as a CFG with the start symbol, S, such that $S \to S_1 S \mid \epsilon$.

4. For each statement below, state whether it is True or False and provide a justification.

   (a) [1 point] Every ambiguous grammar can be converted to an equivalent unambiguous grammar.

   Solution: False. We know that some languages are inherently ambiguous. As such, the grammar that generates them is ambiguous and there is no equivalent unambiguous grammar.

   (b) [1 point] There exist regular languages that are not context-free.

   Solution: False. The class of regular languages is a subset of the class of context-free languages.

   (c) [1 point] There exist context-free languages that are not regular.

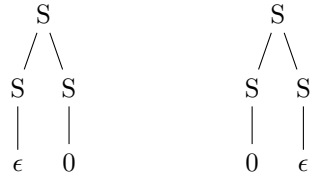
   Solution: True. One witness is $L = \{0^n 1^n \mid n \in \mathbb{Z}^+\}$.

   (d) [1 point] The CFG of a regular language is unambiguous.

   Solution: False. A counterexample follows. Consider the regular language, 0^*, and the following CFG for it.

   $$S \to SS \mid 0 \mid \epsilon$$

   The string, 0, has the following distinct parse trees under this grammar.



   (e) [1 point] Any grammar in CNF is unambiguous.

   Solution: False. We know from (a) above that some ambiguous grammars cannot be converted to an unambiguous grammar. We also know that every grammar can be expressed in CNF. Therefore, not all grammars in CNF are unambiguous.