[10] 1. Answer each question with a **brief** explanation.

(a) If L_1, L_2 are regular languages, is $L_1 \cap L_2$ necessarily regular?

(b) If L_1, L_2 are both not regular languages, is $L_1 \cup L_2$ necessarily not regular?

(b) If $L_1 \cap L_2$ is not regular and L_2 is regular, is L_1 necessarily not regular?

(b) If L_1, L_2 are acceptable, is $L_1 \cap L_2$ necessarily acceptable?

(b) If L is acceptable, is the complement of L necessarily acceptable?

(c) If L is decidable, is the complement of L necessarily decidable?

[10] 2. Write down a DFA for the set of strings over `0,1` that do not contain `110` as a substring. Use the procedure described in class and the text to convert the DFA into an appropriate GNFA and then, by removing states one by one, find a corresponding regular expression.

[10] 3. Show that $\{ww | w \in \{0,1\}^*\}$ is not context-free.

[10] 4. Let G_1 be the grammar `S-> Sa|a`, and let G_2 be the grammar `S-> SS|a`. (Both grammar's describe the language of words consisting of one or more `a`'s.)

(a) How many parse trees are there for `aaa` with G_1, and how many with G_2?

(b) Explain why one of G_1 and G_2 is unambiguous, and the other isn't.

(c) List all rules in Earley's algorithm in bags S_0, S_1, S_2 on input `aaa` for G_1; then do the same for G_2. [Recall that Earley's algorithm adds the rule ϕ`->S`, begins by placing ϕ`->.S 0` into S_0, and has the bag S_i contain all rules obtained after scanning the first i symbols of the word being parsed.]

(d) On input $\mathtt{a}^n$ with n large, explain why Earley's algorithm will be much faster on one of G_1, G_2 than the other.

[10] 5. Recall that

$$A_{\text{TM}} = \{\langle M, w\rangle | M \text{ is a Turing machine and } M \text{ accepts } w\}.$$

Show that A_{TM} is not decidable. [Hint: Suppose R is a Turing machine deciding A_{TM}. Let S be the machine that on input $\langle M\rangle$ runs R on input $\langle M, \langle M\rangle\rangle$ and outputs the opposite of R's answer. What does S do on input $\langle S\rangle$?]

[10] 6. Let 5PLUS be the language of $\langle M\rangle$ such that M accepts at least five strings. Show that 5PLUS is acceptable but not decidable.

[10] 7. Give a reduction to show that 3SAT $\leq_{\text{P}}$ SUBSET-SUM. [Hint: Starting with a 3CNF formula, dedicate one digit (in a suitable base) to each variable and clause in the 3CNF.]

[10] 8. Let DOUBLE-SAT be the set of $\langle\phi\rangle$ such that ϕ is a Boolean formula with at least two satisfying assignments. Show that DOUBLE-SAT is NP-complete.

[10] 9. Let L_n be the language of strings over $\{0,1\}$ of length at least n whose n-th last digit is a 1. In other words, L_n contains precisely those strings of the form $u1w$, where w is a string of length $n-1$ and u is an arbitrary (possibly empty) string.

(a) Write down an NFA accepting L_n with $O(n)$ states.

(b) Show that any DFA accepting L_n has at least 2^n states. [Hint: Show that there is a set of 2^n words that are pairwise distinguishable by L_n, i.e., such that any two of them x, y that are distinct have the property that for some z, exactly one of zx, zy is accepted by L_n.]