Exercise sheet 3

Note: Some new questions have been added in the middle too

1. Design a deterministic finite state automaton over the alphabet $\Sigma = \{0, 1\}$ that will accept a string if and only if it

   a) is the empty string
   b) iS not the empty string
   c) is precisely the string 111
   d) begins with the substring 111
   e) is the empty string or begins with the substring 111
   f) ends with the substring 111
   g) begins with 0 and ends with 1
   h) contains the substring 111

2. Show that if a language L over some alphabet Σ can be recognized by a finite state automaton, then even its complement, L^c (where $L^c := \{s \in \Sigma^* \mid s \notin L\}$), can be recognized by some finite state automaton.

3. We will be discussing this in the lecture, but try to think about it yourself first. Suppose we have an automaton that recognizes a language L_1, and another that recognizes a language L_2.

   a) How will you design an automaton that recognizes $L_1 \cup L_2$?
   b) How will you design an automaton that recognizes $L_1 \cap L_2$?

4. For which of the following languages over the alphabet $\Sigma = \{0, 1\}$ do you feel it is impossible to design a deterministic finite state automaton that recognizes it. For each of the rest, design an automaton to recognize it.

   a) $L = \{ss \mid s \text{ is a string over } \Sigma\}$
   b) The language consisting of precisely those strings that have an even number of 0s and 1s
   c) a language consisting of strings with only 1s and that too a prime number of 1s
   d) The language of all strings with 2 consecutive 0s anywhere in the string
   e) The language consisting of strings with equal number of 0s and 1s
   f) The language consisting of precisesly those strings with more 1s than 0s.

5. Consider the alphabet $\Sigma = \{0, 1, \ldots, n, (,), +, -, *, /\}$. Prove that it is impossible to build a finite state automaton that will reject precisely those strings over Σ that have mis-matched parentheses but accept those with all their parentheses matched. For example, it should reject "(5 + 6 - 9" but should accept "(5+6)-9"

6. Consider the alphabet $\Sigma = \{1\}$ and let $L := \{1^p \mid p \text{ is a prime}\}$. Prove that L cannot be recognized by a finite state automaton.

7. Consider a finite state automaton, M, that recognizes a language, L, over an alphabet, Σ. If we now enlarge the alphabet to Σ', i.e. $\Sigma \subset \Sigma'$, but consider the same set strings as before as our language, how would you modify M to recognize this language? Note that while it accepts exactly the same strings as before, it must reject a lot more strings.

8. By our definition, a finite state automaton *must* read the entire input even if it is already clear that the string must be rejected (for example, if the language includes all strings beginning with a 0). Let us define a "haltable automaton" that can also reject and stop reading the string when there is no point in doing so. Such an automaton may be represented by $(Q, \Sigma, \delta, q_0, F, H)$ where H denotes the subset of "reject and halt states" and the other symbols carry their usual meanings. It works as usual except that if the automaton enters a state in H, it immediately halts and rejects the string reading no more characters. Show that for any such "haltable automaton", the language that it recognizes can also be recognized by a standard deterministic finite state automaton. Therefore, despite the improvement, "haltable automata" recognize no more languages than the usual deterministic finite state automata can.

To be completed