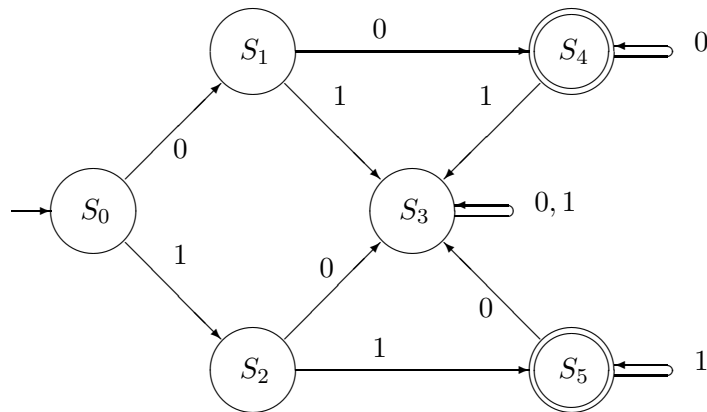


Week 6 Tutorial
Finite State Automata and their Languages

Question 1

The automaton A_1 is specified by the following diagram.



1.1 Review of Notation

1. Transform the graphical representation of A_1 to the form of a 5-tuple, (Σ, S, s_0, F, N) .
2. The next state function is N and so N^* is the eventual state function. What is $N^*(s_2, 00010)$?
3. What language does A_1 accept? Phrase your answer in terms of formulae involving terms such as $L(A)$, $\{w \in \Sigma^* \mid P(w)\}$ and $(w = 0^n \wedge n > 1)$.

1.2 Proofs Involving Automata

1. Consider a statement such as $\forall n \in \mathbb{N}. N^*(s_5, 1^n) = s_5$.
Express it in English. Why is it relevant? How does one prove it?
2. Answer the same questions for the formula $\forall w_1, w_2 \in \Sigma^*. N^*(s_0, w_1 01 w_2) \notin F$

3. If you were doing a proof of the answer you gave in 1.1.3 above, what would be two suitable major goals? (Refer, if necessary to the examples of proof obligations in the lecture notes.)
Give these first in English and then in mathematics similar to that used above (in 1.2.1 and 1.2.2)¹.
4. Break each of these major goals down in suitable subgoals (with an indication of how they would be used).

Question 2

Let A_2 be an automaton with next state function given in the following table. The initial state of the automaton is s_0 and the accepting state is s_f .

State	0	1
s_0	s_1	s_2
s_1	s_3	s_4
s_2	s_4	s_3
s_3	s_5	s_f
s_4	s_f	s_5
s_5	s_5	s_5
s_6	s_6	s_6
s_f	s_6	s_6

Prepare for this exercise by drawing the transition diagram for A_2 and describing the language accepted by A_2 . Do it first in English and then do it using set and predicate calculus notation.

2.1 Proofs of Properties

Prove that your mathematical statement correctly specifies $L(A_2)$. In more detail, you should:

1. Specify the main subgoals
2. Break the subgoals down into easy subsubgoals.
3. Satisfy yourself that these subsubgoals are readily established by standard techniques (arithmetic, induction, tautology checking or whatever).

2.2 Automaton Simplification

1. Give a simpler finite state automaton which accepts the same language.
2. Show that your answer is exactly the one that the standard minimization algorithm would have given.

¹You need not mention the domains for states s , integers n and words w etc. each time they appear.

Question 3

Construct a finite state automaton that accepts the language of binary strings that begin and end with the same bit. Is there any advantage in making it nondeterministic?

Question 4

Virtually every text book that mentions regular languages will cite the language $\{0^n1^n\}$ as an example of a language that is not regular. Moreover, you should remember the way that result is proved (using the pigeon hole principle).

Adapt the standard proof to show that $\{0^i1^j \mid \gcd(i, j) = 1\}$ is not a regular language.