CSE 311 Section 08

Models of Computation

Administrivia

Announcements & Reminders

- HW6 Regrade Requests
  - Submit a regrade request if something was graded incorrectly

- HW7
  - Due **this Friday** 8/15 @11:59pm
  - Late due date **Monday** 8/18 @11:59pm

- Final Exam
  - Thursday 8/21 @ Section
  - Friday 8/22 @ Lecture
  - More information is posted on the Exams page of the course website

Deterministic Finite Automata

Deterministic Finite Automata

- A DFA is a finite-state machine that accepts or rejects a given string of symbols, by running through a state sequence uniquely determined by the string.

- In other words:
  - Our machine is going to get a string as input. It will read one character at a time and update “its state.”
  - At every step, the machine thinks of itself as in one of the (finite number) vertices. When it reads the character, it follows the arrow labeled with that character to its next state.
  - Start at the “start state” (unlabeled, incoming arrow).
  - After you’ve read the last character, accept the string if and only if you’re in a “final state” (double circle).

- Every machine is defined with respect to an alphabet Σ
- Every state has exactly one outgoing edge for every character in Σ
- There is exactly one start state; can have as many accept states (aka final states) as you want – including none.

Problem 1/2 – DFAs

Construct DFAs to recognize each of the following languages.
Let $\Sigma = \{0, 1, 2, 3\}$.

1a) All binary strings.

1b) All strings whose digits sum to an even number.

Let $\Sigma = \{0, 1\}$.

2c) All strings containing an even number of 1's and an odd number of 0's and not containing the substring 10.

Work on these problems with the people around you!

Problem 1 – DFAs, Stage 1

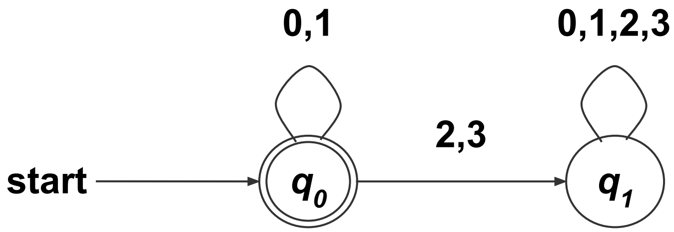
Let $\Sigma = \{0, 1, 2, 3\}$.

a) All binary strings.

Problem 1 – DFAs, Stage 1

Let Σ = {0, 1, 2, 3}.

a) All binary strings.



q_0 : binary strings

q_1 : strings that contain a character which is not 0 or 1

Problem 1 – DFAs, Stage 1

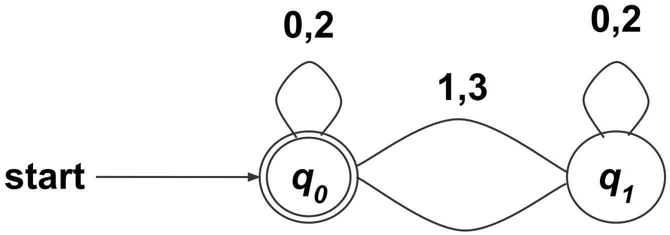
Let $\Sigma = \{0, 1, 2, 3\}$.

b) All strings whose digits sum to an even number.

Problem 1 – DFAs, Stage 1

Let Σ = {0, 1, 2, 3}.

b) All strings whose digits sum to an even number.



q_0: strings whose sum of digits is even
q_1: strings whose sum of digits is odd

Problem 2 – DFAs, Stage 2

Let Σ = {0, 1}.

c) All strings containing an even number of 1’s and an odd number of 0’s and not containing the substring 10.

Problem 2 – DFAs, Stage 2

Let $\Sigma = \{0, 1\}$.

c) All strings containing an even number of 1's and an odd number of 0's and not containing the substring 10.

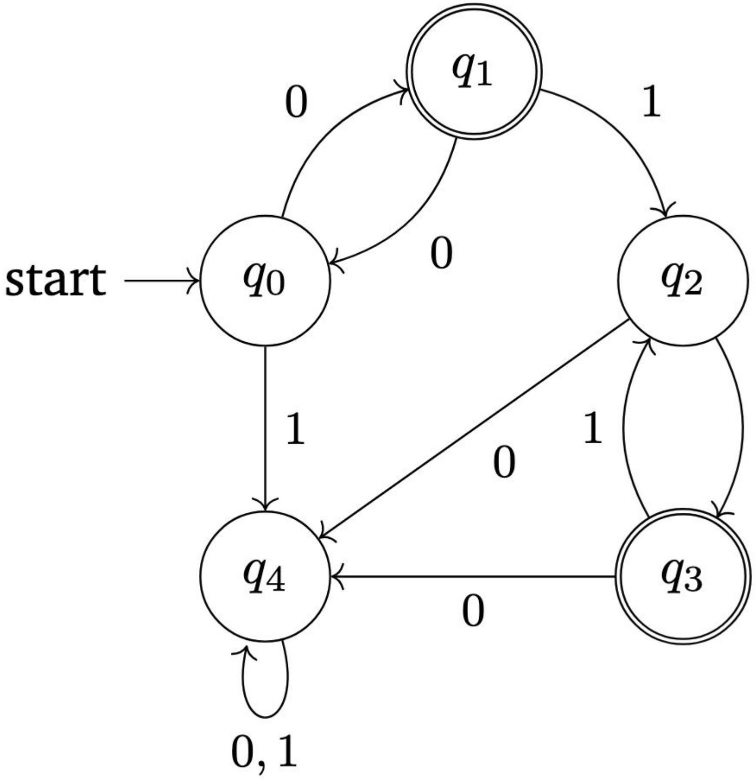
q_0: strings with even number of 1's and even number of 0's (no 10)

q_1: strings with even number of 1's and odd number of 0's (no 10)

q_2: strings with odd number of 1's and odd number of 0's (no 10)

q_3: strings with even number of 1's and odd number of 0's (no 10)

q_4: strings with 10 (can't escape from this state)



Nondeterministic Finite Automata

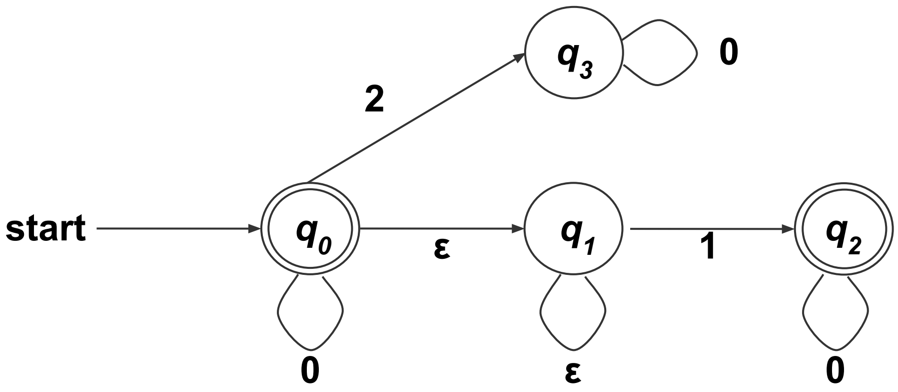
Nondeterministic Finite Automata

- Similar to DFAs, but with less restrictions.
  - From a given state, we'll allow any number of outgoing edges labeled with a given character. (In a DFA, we have only 1 outgoing edge labeled with each character).
  - The machine can follow any of them.
  - We'll have edges labeled with "ε" – the machine (optionally) can follow one of those without reading another character from the input.
  - If we "get stuck" i.e. the next character is a and there's no transition leaving our state labeled a, the computation dies.

- An NFA still has exactly one start state and any number of final states.
- The NFA accepts x if there is some path from a start state to a final state labeled with x.
- From a state, you can have 0,1, or many outgoing arrows labeled with a single character. You can choose any of them to build the required path.

Problem 3 – NFAs

a) What language does the following NFA accept?

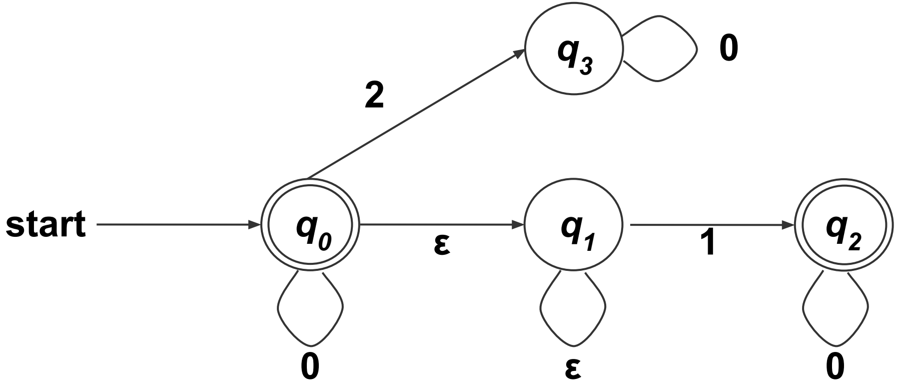


b) Create an NFA for the language “all binary strings that have a 1 as one of the last three digits”.

Work on this problem with the people around you!

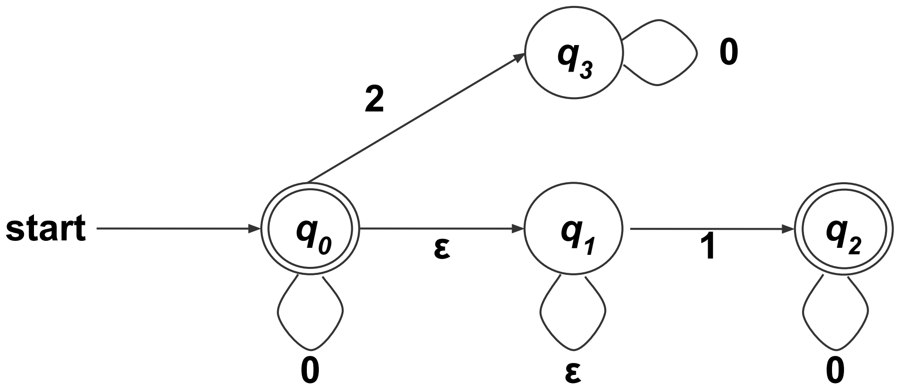
Problem 3 – NFAs

a) What language does the following NFA accept?



Problem 3 – NFAs

a) What language does the following NFA accept?



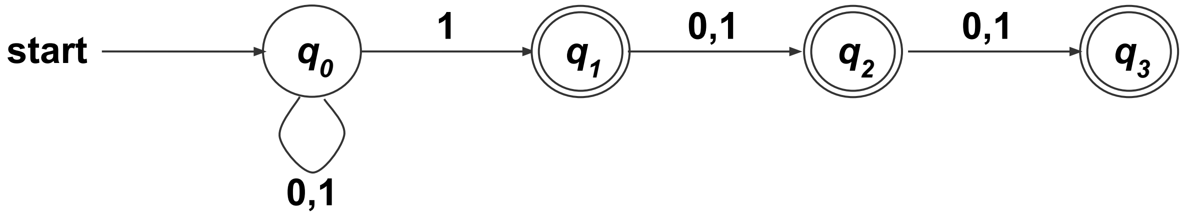
All strings of only 0's and 1's, not containing more than one 1.

Problem 3 – NFAs

b) Create an NFA for the language “all binary strings that have a 1 as one of the last three digits”.

Problem 3 – NFAs

b) Create an NFA for the language “all binary strings that have a 1 as one of the last three digits”.



Irregularity

Irregularity Template

Claim: L is an irregular language.

Proof: Suppose, for the sake of contradiction, that L is regular. Then there is a DFA M such that M accepts exactly L.

Let S = [TODO] (S is an infinite set of strings)
Because the DFA is finite, there are two (different) strings x, y in S such that x and y go to the same state when read by M. [TODO] (We don't get to choose x, y, but we can describe them based on that set S we just defined)

Consider the string z = [TODO] (We do get to choose z depending on x, y)

Since x, y led to the same state and M is deterministic, xz and yz will also lead to the same state q in M. Observe that xz = [TODO], so $xz \in L$ but yz = [TODO], so $yz \notin L$. Since q is can be only one of an accept or reject state, M does not actually recognize L. That's a contradiction!

Therefore, L is an irregular language.

Irregularity Example

Claim: $\{0^k 1^k : k \geq 0\}$ is an irregular language.

Proof: Suppose, for the sake of contradiction, that $L = \{0^k 1^k : k \geq 0\}$ is regular. Then there is a DFA M such that M accepts exactly L.

Let $S = \{0^k : k \geq 0\}$
Because the DFA is finite, there are two (different) strings x, y in S such that x and y go to the same state when read by M. Since both are in S, $x = 0^a$ for some integer $a \geq 0$, and $y = 0^b$ for some integer $b \geq 0$, with $a \neq b$.

Consider the string $z = 1^{a}$.

Since x, y led to the same state and M is deterministic, xz and yz will also lead to the same state q in M. Observe that $xz = 0^a 1^a$, so $xz \in L$ but $yz = 0^{b} 1^a$, so $yz \notin L$. Since q is can be only one of an accept or reject state, M does not actually recognize L. That's a contradiction!

Therefore, L is an irregular language.

Problem 5 – Irregularity

a) Let $\Sigma = \{0, 1\}$. Prove that $\{0^n1^n0^n : n \geq 0\}$ is not regular.

b) Let $\Sigma = \{0, 1, 2\}$. Prove that $\{0^n(12)^m : n \geq m \geq 0\}$ is not regular.

Work on this problem with the people around you!

Problem 5 – Irregularity

(a) Let $\Sigma = \{0, 1\}$. Prove that $\{0^n1^n0^n : n \geq 0\}$ is not regul

Claim: $\{0^n1^n0^n : n \geq 0\}$ is an irregular language.

Proof: Suppose, for the sake of contradiction, that $L = \{0^n1^n0^n : n \geq 0\}$ is regular. Then there is a DFA M such that M accepts exactly L.

Let S = [TODO]
Because the DFA is finite, there are two (different) strings x, y in S such that x and y go to the same state when read by M. [TODO] .

Consider the string z = [TODO] .

Since x, y led to the same state and M is deterministic, xz and yz will also lead to the same state q in M. Observe that xz = [TODO] , so $xz \in L$ but yz = [TODO] , so $yz \notin L$. Since q is can be only one of an accept or reject state, M does not actually recognize L. That's a contradiction!

Therefore, L is an irregular language.

Problem 5 – Irregularity

(a) Let $\Sigma = \{0, 1\}$. Prove that $\{0^n1^n0^n : n \geq 0\}$ is not regul

Claim: $\{0^n1^n0^n : n \geq 0\}$ is an irregular language.

Proof: Suppose, for the sake of contradiction, that $L = \{0^n1^n0^n : n \geq 0\}$ is regular. Then there is a DFA M such that M accepts exactly L.

Let $S = \{0^n1^n : n \geq 0\}$
Because the DFA is finite, there are two (different) strings x, y in S such that x and y go to the same state when read by M. [TODO] .

Consider the string z = [TODO] .

Since x, y led to the same state and M is deterministic, xz and yz will also lead to the same state q in M. Observe that xz = [TODO] , so $xz \in L$ but yz = [TODO] , so $yz \notin L$. Since q is can be only one of an accept or reject state, M does not actually recognize L. That's a contradiction!

Therefore, L is an irregular language.

Problem 5 – Irregularity

(a) Let $\Sigma = \{0, 1\}$. Prove that $\{0^n1^n0^n : n \geq 0\}$ is not regul

Claim: $\{0^n1^n0^n : n \geq 0\}$ is an irregular language.

Proof: Suppose, for the sake of contradiction, that $L = \{0^n1^n0^n : n \geq 0\}$ is regular. Then there is a DFA M such that M accepts exactly L.

Let $S = \{0^n1^n : n \geq 0\}$
Because the DFA is finite, there are two (different) strings x, y in S such that x and y go to the same state when read by M. Since both are in S, $x = 0^a1^a$ for some integer $a \geq 0$, and $y = 0^b1^b$ for some integer $b \geq 0$, with $a \neq b$.

Consider the string z = [TODO] .

Since x, y led to the same state and M is deterministic, xz and yz will also lead to the same state q in M. Observe that xz = [TODO] , so $xz \in L$ but yz = [TODO] , so $yz \notin L$. Since q is can be only one of an accept or reject state, M does not actually recognize L. That's a contradiction!

Therefore, L is an irregular language.

Problem 5 – Irregularity

(a) Let $\Sigma = \{0, 1\}$. Prove that $\{0^n1^n0^n : n \geq 0\}$ is not regul

Claim: $\{0^n1^n0^n : n \geq 0\}$ is an irregular language.

Proof: Suppose, for the sake of contradiction, that $L = \{0^n1^n0^n : n \geq 0\}$ is regular. Then there is a DFA M such that M accepts exactly L.

Let $S = \{0^n1^n : n \geq 0\}$
Because the DFA is finite, there are two (different) strings x, y in S such that x and y go to the same state when read by M. Since both are in S, $x = 0^a1^a$ for some integer $a \geq 0$, and $y = 0^b1^b$ for some integer $b \geq 0$, with $a \neq b$.

Consider the string $z = 0^a$.

Since x, y led to the same state and M is deterministic, xz and yz will also lead to the same state q in M. Observe that xz = [TODO] , so $xz \in L$ but yz = [TODO] , so $yz \notin L$. Since q is can be only one of an accept or reject state, M does not actually recognize L. That's a contradiction!

Therefore, L is an irregular language.

Problem 5 – Irregularity

(a) Let $\Sigma = \{0, 1\}$. Prove that $\{0^n1^n0^n : n \geq 0\}$ is not regul

Claim: $\{0^n1^n0^n : n \geq 0\}$ is an irregular language.

Proof: Suppose, for the sake of contradiction, that $L = \{0^n1^n0^n : n \geq 0\}$ is regular. Then there is a DFA M such that M accepts exactly L.

Let $S = \{0^n1^n : n \geq 0\}$
Because the DFA is finite, there are two (different) strings x, y in S such that x and y go to the same state when read by M. Since both are in S, $x = 0^a1^a$ for some integer $a \geq 0$, and $y = 0^b1^b$ for some integer $b \geq 0$, with $a \neq b$.

Consider the string $z = 0^a$.

Since x, y led to the same state and M is deterministic, xz and yz will also lead to the same state q in M. Observe that $xz = 0^a1^a0^a$, so $xz \in L$ but $yz = 0^b1^b0^a$, so $yz \notin L$. Since q is can be only one of an accept or reject state, M does not actually recognize L. That's a contradiction!

Therefore, L is an irregular language.

Problem 5 – Irregularity

(b) Let $\Sigma = \{0, 1, 2\}$. Prove that $\{0^n(12)^m : n \geq m \geq 0\}$ is not regular.

Claim: $\{0^n(12)^m : n \geq m \geq 0\}$ is an irregular language.

Proof: Suppose, for the sake of contradiction, that $L = \{0^n(12)^m : n \geq m \geq 0\}$ is regular. Then there is a DFA M such that M accepts exactly L.

Let S = [TODO]
Because the DFA is finite, there are two (different) strings x, y in S such that x and y go to the same state when read by M. [TODO] .

Consider the string z = [TODO] .

Since x, y led to the same state and M is deterministic, xz and yz will also lead to the same state q in M. Observe that xz = [TODO] , so $xz \in L$ but yz = [TODO] , so $yz \notin L$. Since q is can be only one of an accept or reject state, M does not actually recognize L. That's a contradiction!

Therefore, L is an irregular language.

Problem 5 – Irregularity

(b) Let $\Sigma = \{0, 1, 2\}$. Prove that $\{0^n(12)^m : n \geq m \geq 0\}$ is not regular.

Claim: $\{0^n(12)^m : n \geq m \geq 0\}$ is an irregular language.

Proof: Suppose, for the sake of contradiction, that $L = \{0^n(12)^m : n \geq m \geq 0\}$ is regular. Then there is a DFA M such that M accepts exactly L.

Let S = $\{0^n : n \geq 0\}$
Because the DFA is finite, there are two (different) strings x, y in S such that x and y go to the same state when read by M. [TODO]

Consider the string z = [TODO] .

Since x, y led to the same state and M is deterministic, xz and yz will also lead to the same state q in M. Observe that xz = [TODO] , so $xz \in L$ but yz = [TODO] , so $yz \notin L$. Since q is can be only one of an accept or reject state, M does not actually recognize L. That's a contradiction!

Therefore, L is an irregular language.

Problem 5 – Irregularity

(b) Let $\Sigma = \{0, 1, 2\}$. Prove that $\{0^n(12)^m : n \geq m \geq 0\}$ is not regular.

Claim: $\{0^n(12)^m : n \geq m \geq 0\}$ is an irregular language.

Proof: Suppose, for the sake of contradiction, that $L = \{0^n(12)^m : n \geq m \geq 0\}$ is regular. Then there is a DFA M such that M accepts exactly L.

Let $S = \{0^n : n \geq 0\}$
Because the DFA is finite, there are two (different) strings x, y in S such that x and y go to the same state when read by M. Since both are in S, $x = 0^a$ for some integer $a \geq 0$, and $y = 0^b$ for some integer $b \geq 0$. Assume without loss of generality that $a > b$. (Or go by cases).

Consider the string z = [TODO] .

Since x, y led to the same state and M is deterministic, xz and yz will also lead to the same state q in M. Observe that xz = [TODO] , so $xz \in L$ but yz = [TODO] , so $yz \notin L$. Since q is can be only one of an accept or reject state, M does not actually recognize L. That's a contradiction!

Therefore, L is an irregular language.

Problem 5 – Irregularity

(b) Let $\Sigma = \{0, 1, 2\}$. Prove that $\{0^n(12)^m : n \geq m \geq 0\}$ is not regular.

Claim: $\{0^n(12)^m : n \geq m \geq 0\}$ is an irregular language.

Proof: Suppose, for the sake of contradiction, that $L = \{0^n(12)^m : n \geq m \geq 0\}$ is regular. Then there is a DFA M such that M accepts exactly L.

Let S = $\{0^n : n \geq 0\}$
Because the DFA is finite, there are two (different) strings x, y in S such that x and y go to the same state when read by M. Since both are in S, $x = 0^a$ for some integer $a \geq 0$, and $y = 0^b$ for some integer $b \geq 0$, Assume without loss of generality that $a > b$. (Or go by cases).

Consider the string z = $(12)^a$.

Since x, y led to the same state and M is deterministic, xz and yz will also lead to the same state q in M. Observe that xz = [TODO] , so $xz \in L$ but yz = [TODO] , so $yz \notin L$. Since q is can be only one of an accept or reject state, M does not actually recognize L. That's a contradiction!

Therefore, L is an irregular language.

Problem 5 – Irregularity

(b) Let $\Sigma = \{0, 1, 2\}$. Prove that $\{0^n(12)^m : n \geq m \geq 0\}$ is not regular.

Claim: $\{0^n(12)^m : n \geq m \geq 0\}$ is an irregular language.

Proof: Suppose, for the sake of contradiction, that $L = \{0^n(12)^m : n \geq m \geq 0\}$ is regular. Then there is a DFA M such that M accepts exactly L.

Let $S = \{0^n : n \geq 0\}$
Because the DFA is finite, there are two (different) strings x, y in S such that x and y go to the same state when read by M. Since both are in S, $x = 0^a$ for some integer $a \geq 0$, and $y = 0^b$ for some integer $b \geq 0$. Assume without loss of generality that $a > b$. (Or go by cases).

Consider the string $z = (12)^a$.

Since x, y led to the same state and M is deterministic, xz and yz will also lead to the same state q in M. Observe that $xz = 0^a(12)^a$, so $xz \in L$ but $yz = 0^b(12)^a$, so $yz \notin L$. Since q is can be only one of an accept or reject state, M does not actually recognize L. That's a contradiction!

Therefore, L is an irregular language.

All The Models

Problem 4 – All The Models

Construct a valid regular expression, CFG, and DFA for the following languages.

a) All strings whose base-6 representation is divisible by 3 (leading zeros are ok). Let $\Sigma = \{0, 1, 2, 3, 4, 5\}$.

a) All binary strings of 0s capped by a 1 on either side

Work on this problem with the people around you!

Problem 4 – All The Models

a) All strings whose base-6 representation is divisible by 3 (leading zeros are ok). Let $\Sigma = \{0, 1, 2, 3, 4, 5\}$.

Problem 4 – All The Models

a) All strings whose base-6 representation is divisible by 3 (leading zeros are ok). Let $\Sigma = \{0, 1, 2, 3, 4, 5\}$.

Regular Expression: $(0 \cup 1 \cup 2 \cup 3 \cup 4 \cup 5)^*(0 \cup 3)$

Problem 4 – All The Models

a) All strings whose base-6 representation is divisible by 3 (leading zeros are ok). Let $\Sigma = \{0, 1, 2, 3, 4, 5\}$.

Regular Expression: $(0 \cup 1 \cup 2 \cup 3 \cup 4 \cup 5)^*(0 \cup 3)$

CFG:

$$\mathbf{S} \to \mathbf{T}0 \mid \mathbf{T}3$$
$$\mathbf{T} \to 0\mathbf{T} \mid 1\mathbf{T} \mid 2\mathbf{T} \mid 3\mathbf{T} \mid 4\mathbf{T} \mid 5\mathbf{T} \mid \varepsilon$$

Problem 4 – All The Models

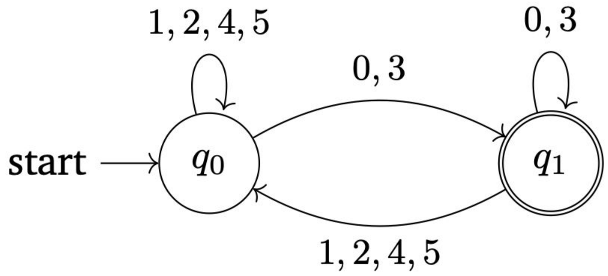
a) All strings whose base-6 representation is divisible by 3 (leading zeros are ok). Let $\Sigma = \{0, 1, 2, 3, 4, 5\}$.

Regular Expression: $(0 \cup 1 \cup 2 \cup 3 \cup 4 \cup 5)^*(0 \cup 3)$

CFG:

$$\mathbf{S} \rightarrow \mathbf{T}0 \mid \mathbf{T3}$$
$$\mathbf{T} \rightarrow 0\mathbf{T} \mid \mathbf{1T} \mid \mathbf{2T} \mid \mathbf{3T} \mid \mathbf{4T} \mid \mathbf{5T} \mid \varepsilon$$

DFA:



Problem 4 – All The Models

b) All binary strings of 0s capped by a 1 on either side

Problem 4 – All The Models

b) All binary strings of 0s capped by a 1 on either side

Binary strings, therefore $\Sigma = \{0, 1\}$.

Problem 4 – All The Models

b) All binary strings of 0s capped by a 1 on either side

Binary strings, therefore Σ = {0, 1}.

Regular Expression: $1(0)^*1$

Problem 4 – All The Models

b) All binary strings of 0s capped by a 1 on either side

Binary strings, therefore $\Sigma = \{0, 1\}$.

Regular Expression: $1(0)^*1$

Context-Free Grammar:

$$\mathbf{S} \rightarrow 1\mathbf{T}1$$
$$\mathbf{T} \rightarrow 0\mathbf{T} \mid \varepsilon$$

Problem 4 – All The Models

b) All binary strings of 0s capped by a 1 on either side

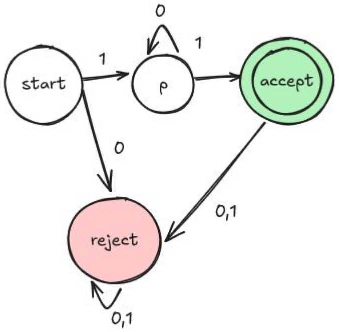
Binary strings, therefore Σ = {0, 1}.

Regular Expression: $1(0)^*1$

Context-Free Grammar:

$$\mathbf{S} \rightarrow 1\mathbf{T}1$$
$$\mathbf{T} \rightarrow 0\mathbf{T} \mid \varepsilon$$

Deterministic Finite Automata:



Problem 4 – All The Models

b) All binary strings of 0s capped by a 1 on either side

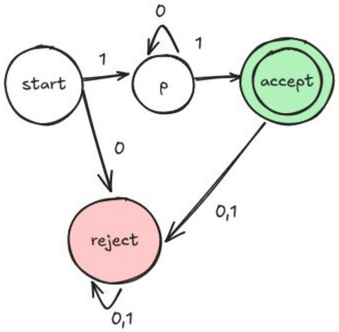
Binary strings, therefore Σ = {0, 1}.

Regular Expression: $1(0)^*1$

Context-Free Grammar:

$$\mathbf{S} \rightarrow 1\mathbf{T}1$$
$$\mathbf{T} \rightarrow 0\mathbf{T} \mid \varepsilon$$

Deterministic Finite Automata:



What if we made the problem more complex?

All binary strings of 0s capped by a **111000111** on either side

Problem 4 – All The Models

b) All binary strings of 0s capped by a **111000111** on either side

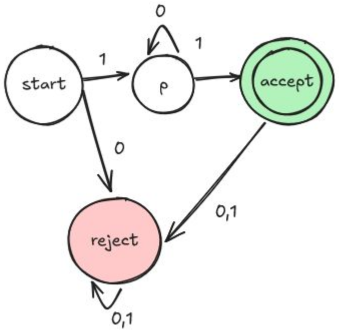
Binary strings, therefore $\Sigma = \{0, 1\}$.

Regular Expression: $1(0)^*1$

Context-Free Grammar:

$$\mathbf{S} \to 1\mathbf{T}1$$
$$\mathbf{T} \to 0\mathbf{T} \mid \varepsilon$$

Deterministic Finite Automata:



Problem 4 – All The Models

b) All binary strings of 0s capped by a **111000111** on either side

Binary strings, therefore Σ = {0, 1}.

Regular Expression:

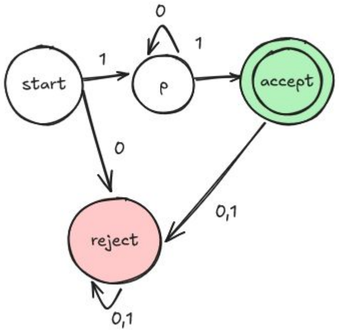
$111000111\,(0)^*\,111000111$

Context-Free Grammar:

$S \to 111000111\ T\ 111000111$

$T \to 0T \mid \varepsilon$

Deterministic Finite Automata:

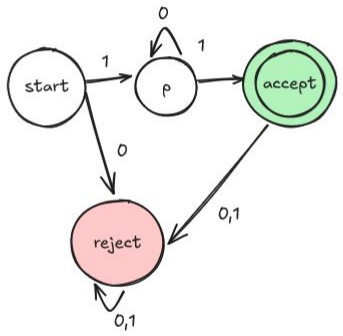


Just like RE and CFG, what structure can we replace with complexity?

Problem 4 – All The Models

b) All binary strings of 0s capped by a **111000111** on either side

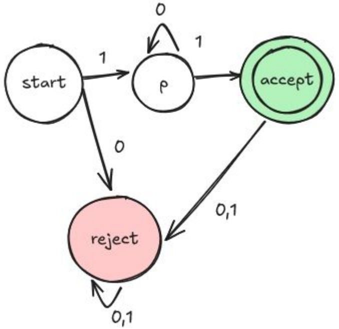
Deterministic Finite Automata:

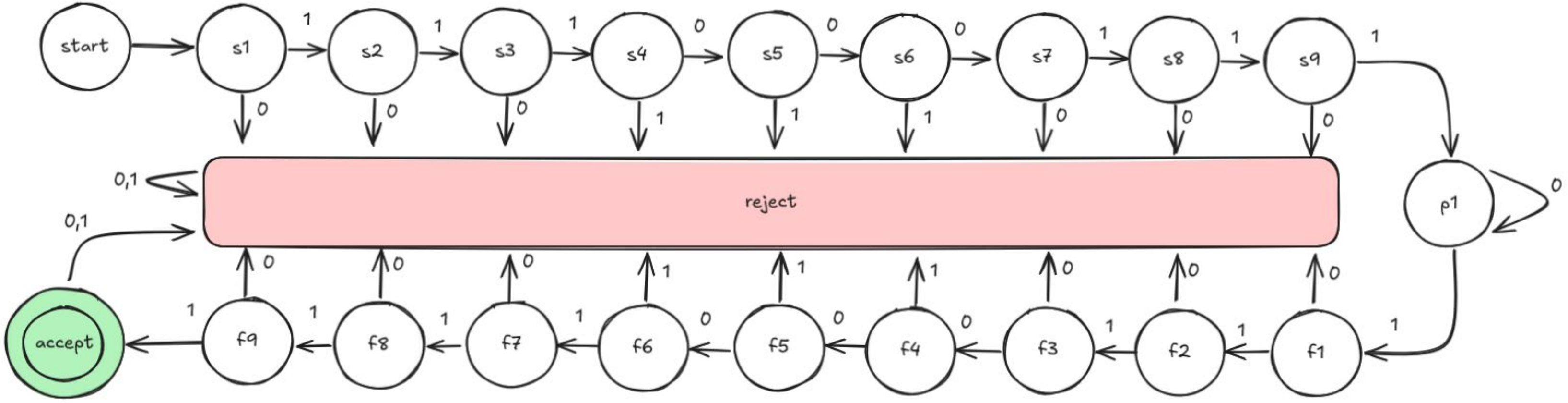


Problem 4 – All The Models

b) All binary strings of 0s capped by a **111000111** on either side

Deterministic Finite Automata:





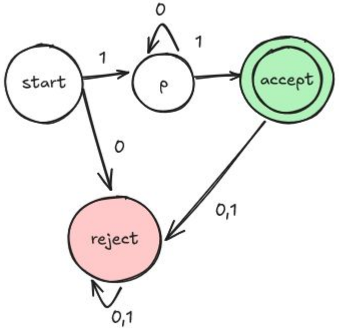
Problem 4 – All The Models

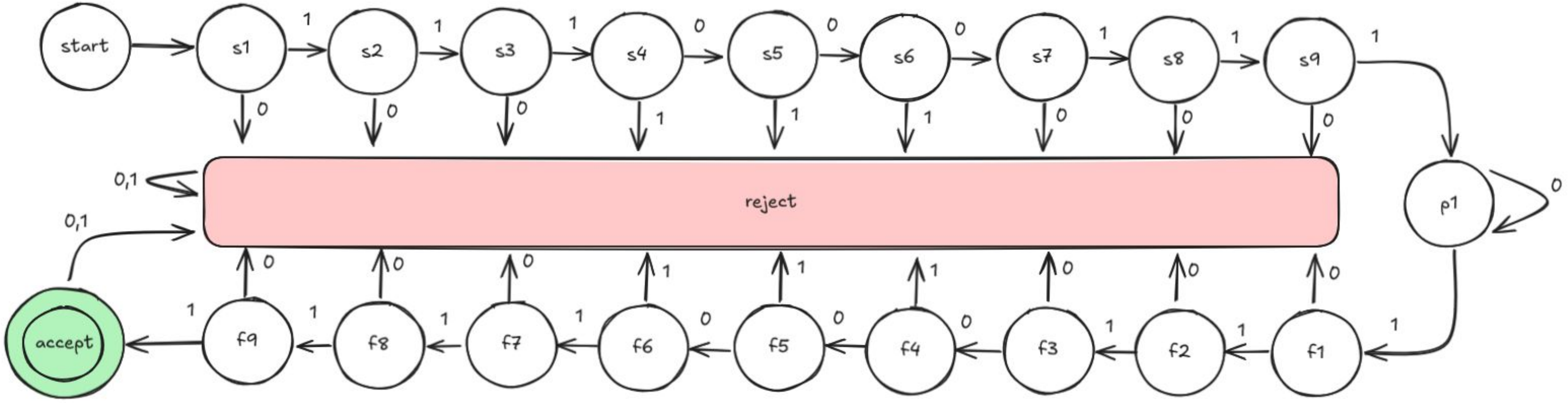
b) All binary strings of 0s capped by a **111000111** on either side

Deterministic Finite Automata:

What structure can we replace with complexity?

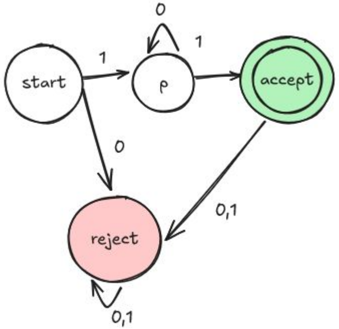
At most n 0s?



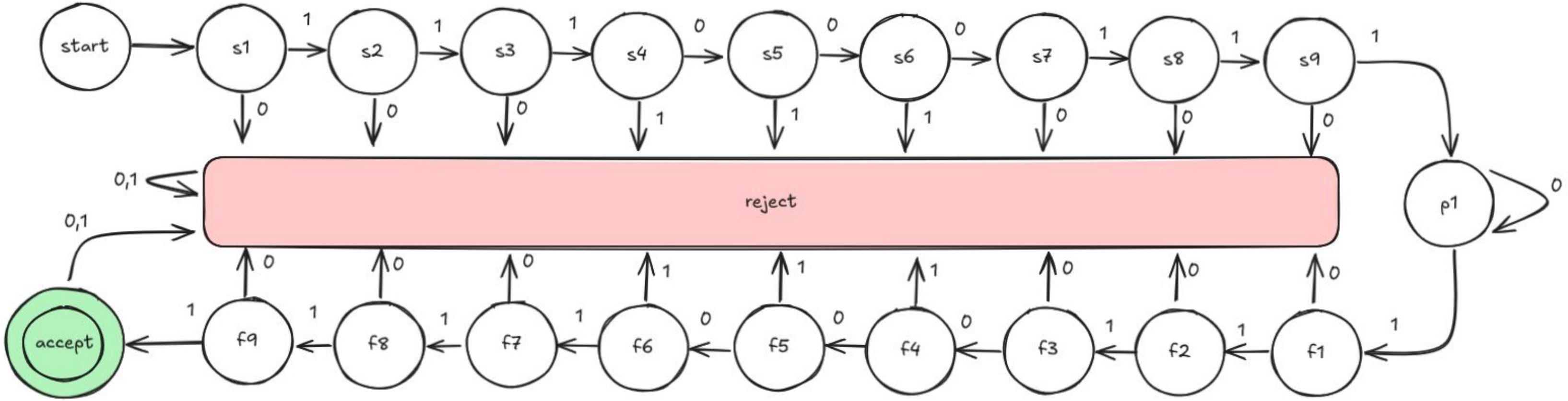


Problem 4 – All The Models

b) All binary strings of 0s capped by a **111000111** on either side



Takeaway: start with top level structure, fill in finegrained details (like circuits)



That’s All, Folks!

Thanks for coming to section this week!
Any questions?