Final Review Session

CSE 311 - WI 2022
Alice Wang
Allie Pfleger
Muru Zhang

Warm-up: Predicate Logic

Express each of these system specifications using predicates, quantifiers, and logical connectives. For some of these problems, more than one translation will be reasonable depending on your choice of predicates.

(a) Every user has access to an electronic mailbox

(b) The system mailbox can be accessed by everyone in the group if the file system is locked.

(c) The firewall is in a diagnostic state only if the proxy server is in a diagnostic state.

(d) At least one router is functioning normally if the throughput is between 100kbps and 500 kbps and the proxy server is not in diagnostic mode.

Warm-up: Predicate Logic Solutions

(a) Every user has access to an electronic mailbox.

Let the domain be users and mailboxes. Let User(x) be "x is a user", let Mailbox(y) be "y is a mailbox", and let Access(x, y) be "x has access to y".

$$\forall x\, (\text{User}(x) \rightarrow (\exists y\, (\text{Mailbox}(y) \land \text{Access}(x, y))))$$

(b) The system mailbox can be accessed by everyone in the group if the file system is locked.

Warm-up: Predicate Logic Solutions

(a) Every user has access to an electronic mailbox.

Let the domain be users and mailboxes. Let User(x) be "x is a user", let Mailbox(y) be "y is a mailbox", and let Access(x, y) be "x has access to y".

$$\forall x\,(\text{User}(x) \rightarrow (\exists y\,(\text{Mailbox}(y) \wedge \text{Access}(x, y))))$$

(b) The system mailbox can be accessed by everyone in the group if the file system is locked.

Solution 1: Let the domain be people in the group. Let CanAccessSM(x) be "x has access to the system mailbox". Let p be the proposition "the file system is locked."

$$p \rightarrow \forall x\,\text{CanAccessSM}(x).$$

Solution2: Let the domain be people and mailboxes and use Access(x, y) as defined in the solution to part (a), and then also add InGroup(x) for "x is in the group", and let SystemMailBox be the name for the system mailbox.

$$\text{FileSystemLocked} \rightarrow \forall x\,(\text{InGroup}(x) \rightarrow \text{Access}(x, \text{SystemMailBox})).$$

Warm-up: Predicate Logic Solutions

(c) The firewall is in a diagnostic state only if the proxy server is in a diagnostic state.

Let the domain be all applications. Let Firewall(x) be "x is the firewall", and let ProxyServer(x) be "x is the proxy server." Let Diagnostic(x) be "x is in a diagnostic state".

$$\forall x\, \forall y\, ((\text{Firewall}(x) \land \text{Diagnostic}(x)) \to (\text{ProxyServer}(y) \to \text{Diagnostic}(y))$$

(d) At least one router is functioning normally if the throughput is between 100kbps and 500 kbps and the proxy server is not in diagnostic mode.

Warm-up: Predicate Logic Solutions

(c) The firewall is in a diagnostic state only if the proxy server is in a diagnostic state.

Let the domain be all applications. Let Firewall(x) be "x is the firewall", and let ProxyServer(x) be "x is the proxy server." Let Diagnostic(x) be "x is in a diagnostic state".

$$\forall x\, \forall y\, ((\text{Firewall}(x) \land \text{Diagnostic}(x)) \rightarrow (\text{ProxyServer}(y) \rightarrow \text{Diagnostic}(y))$$

(d) At least one router is functioning normally if the throughput is between 100kbps and 500 kbps and the proxy server is not in diagnostic mode.

Let the domain be all applications and routers. Let Router(x) be "x is a router", and let ProxyServer(x) be "x is the proxy server." Let Diagnostic(x) be "x is in a diagnostic state". Let p be "the throughput is between 100kbps and 500 kbps". Let Functioning(y) be "y is functioning normally".

$$p \land \forall x\, (\neg\, \text{ProxyServer}(x) \lor \neg\, \text{Diagnostic}(x))) \rightarrow \exists y\, (\text{Router}(y) \land \text{Functioning}(y))$$

$\neg(PS(x) \land D(x))$

Practice Final: 1. Regularly Irregular

Let $\Sigma = \{0, 1\}$. Prove that the language $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ is irregular.

Practice Final: 1. Regularly Irregular Solution

Let $\Sigma = \{0, 1\}$. Prove that the language $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ is irregular.

Suppose, for the sake of contradiction, that $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ 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.

Practice Final: 1. Regularly Irregular Solution

Let $\Sigma = \{0, 1\}$. Prove that the language $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ is irregular.

Suppose, for the sake of contradiction, that $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ 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.

Practice Final: 1. Regularly Irregular Solution

Let $\Sigma = \{0, 1\}$. Prove that the language $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ is irregular.

Suppose, for the sake of contradiction, that $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ 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 $b \geq 0$, with $a < b$.

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.

Practice Final: 1. Regularly Irregular Solution

Let $\Sigma = \{0, 1\}$. Prove that the language $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ is irregular.

Suppose, for the sake of contradiction, that $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ 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 $b \geq 0$, with $a < b$.

Consider the string $z = 1^b$.

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.

Practice Final: 1. Regularly Irregular Solution

Let $\Sigma = \{0, 1\}$. Prove that the language $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ is irregular.

Suppose, for the sake of contradiction, that $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ 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 $b \geq 0$, with $a < b$.

Consider the string $z = 1^b$.

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 $xz = 0^a 1^b$, 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.

Practice Final: 1. Regularly Irregular Solution

Let $\Sigma = \{0, 1\}$. Prove that the language $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ is irregular.

Suppose, for the sake of contradiction, that $L = \{x \in \Sigma^* : \#_0(x) < \#_1(x)\}$ 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 $b \geq 0$, with $a < b$.

Consider the string $z = 1^b$.

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 $xz = 0^a1^b$, so $xz \in L$ but $yz = 0^b1^b$, 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.

Practice Final: 2. Recurrences, Recurrences

Define

$$T(n) = \begin{cases} n & \text{if } n = 0, 1 \\ 4T\left(\lfloor \frac{n}{2} \rfloor\right) + n & \text{otherwise} \end{cases}$$

Prove that $T(n) \leq n^3$ for $n \geq 3$

Practice Final: 2. Recurrences, Recurrences Solution

We go by strong induction on n. Let P(n) be "$T(n) \leq n^3$" for $n \geq 3$.

<u>Base Cases.</u>

<u>Induction Hypothesis.</u>

<u>Induction Step.</u>

<u>Conclusion</u>.

Practice Final: 2. Recurrences, Recurrences Solution

We go by strong induction on n. Let P(n) be "$T(n) \leq n^3$" for $n \geq 3$.

Base Cases. When n = 3:

When n = 4:

When n = 5:

Induction Hypothesis.

Induction Step.

Conclusion.

Practice Final: 2. Recurrences, Recurrences Solution

We go by strong induction on n. Let P(n) be "$T(n) \leq n^3$" for $n \geq 3$.

Base Cases. When n = 3: $T(3) = 4T\left(\left\lfloor \frac{3}{2} \right\rfloor\right) + 3 = 4T(1) + 3 = 7 \leq 27 = 3^3$.

When n = 4: $T(4) = 4T\left(\left\lfloor \frac{4}{2} \right\rfloor\right) + 4 = 4T(2) + 4 = 28 \leq 64 = 4^3$.

When n = 5: $T(5) = 4T\left(\left\lfloor \frac{5}{2} \right\rfloor\right) + 5 = 4T(2) + 5 = 29 \leq 4^4$.

Induction Hypothesis.

Induction Step.

Conclusion.

Practice Final: 2. Recurrences, Recurrences Solution

We go by strong induction on n. Let P(n) be "$T(n) \leq n^3$" for $n \geq 3$.

Base Cases. When n = 3: $T(3) = 4T\left(\left\lfloor \frac{3}{2} \right\rfloor\right) + 3 = 4T(1) + 3 = 7 \leq 27 = 3^3$.

When n = 4: $T(4) = 4T\left(\left\lfloor \frac{4}{2} \right\rfloor\right) + 4 = 4T(2) + 4 = 28 \leq 64 = 4^3$.

When n = 5: $T(5) = 4T\left(\left\lfloor \frac{5}{2} \right\rfloor\right) + 5 = 4T(2) + 5 = 29 \leq 4^4$.

Induction Hypothesis. Suppose $P(3) \wedge P(4) \wedge \cdots \wedge P(k)$ for some $k \geq 5$.

Induction Step.

Conclusion.

Practice Final: 2. Recurrences, Recurrences Solution

We go by strong induction on n. Let P(n) be "T(n) ≤ n^3" for n ≥ 3.

Base Cases. When n = 3: $T(3) = 4T\left(\left\lfloor \frac{3}{2} \right\rfloor\right) + 3 = 4T(1) + 3 = 7 \leq 27 = 3^3$.

When n = 4: $T(4) = 4T\left(\left\lfloor \frac{4}{2} \right\rfloor\right) + 4 = 4T(2) + 4 = 28 \leq 64 = 4^3$.

When n = 5: $T(5) = 4T\left(\left\lfloor \frac{5}{2} \right\rfloor\right) + 5 = 4T(2) + 5 = 29 \leq 4^4$.

Induction Hypothesis. Suppose P(3) ∧ P(4) ∧ · · · ∧ P(k) for some k ≥ 5.

$T(x) \leq k^3$

Induction Step.

$$
\begin{aligned}
T(k+1) &= 4T\left(\left\lfloor \frac{k+1}{2} \right\rfloor\right) + k + 1, && \text{because } k+1 \geq 2. \\
&\leq 4\left(\left\lfloor \frac{k+1}{2} \right\rfloor\right)^3 + k + 1, && \text{by IH.} \\
&\leq 4\left(\frac{k+1}{2}\right)^3 + k + 1, && \text{by def of floor.} \\
&= 4\left(\frac{(k+1)^3}{2^3}\right) + k + 1, && \text{by algebra.} \\
&= \frac{(k+1)^3}{2} + k + 1, && \text{by algebra.} \\
&= \frac{(k+1)((k+1)^2 + 2)}{2}, && \text{by algebra.} \\
&\leq \frac{(k+1)((k+1)^2 + (k+1)^2)}{2}, && \text{because } (k+1)^2 \geq 2. \\
&= (k+1)^3, && \text{by algebra}
\end{aligned}
$$

Conclusion.

Practice Final: 2. Recurrences, Recurrences Solution

We go by strong induction on n. Let P(n) be "T(n) ≤ n^3" for n ≥ 3.

Base Cases. When n = 3: $T(3) = 4T\left(\left\lfloor \frac{3}{2} \right\rfloor\right) + 3 = 4T\left(1\right) + 3 = 7 \leq 27 = 3^3$.

When n = 4: $T(4) = 4T\left(\left\lfloor \frac{4}{2} \right\rfloor\right) + 4 = 4T\left(2\right) + 4 = 28 \leq 64 = 4^3$.

When n = 5: $T(5) = 4T\left(\left\lfloor \frac{5}{2} \right\rfloor\right) + 5 = 4T\left(2\right) + 5 = 29 \leq 4^4$.

Induction Hypothesis. Suppose P(3) ∧ P(4) ∧ · · · ∧ P(k) for some k ≥ 5.

Induction Step.

$$
\begin{aligned}
T(k+1) &= 4T\left(\left\lfloor \frac{k+1}{2} \right\rfloor\right) + k + 1, && \text{because } k+1 \geq 2. \\
&\leq 4\left(\left\lfloor \frac{k+1}{2} \right\rfloor\right)^3 + k + 1, && \text{by IH.} \\
&\leq 4\left(\frac{k+1}{2}\right)^3 + k + 1, && \text{by def of floor.} \\
&= 4\left(\frac{(k+1)^3}{2^3}\right) + k + 1, && \text{by algebra.} \\
&= \frac{(k+1)^3}{2} + k + 1, && \text{by algebra.} \\
&= \frac{(k+1)((k+1)^2 + 2)}{2}, && \text{by algebra.} \\
&\leq \frac{(k+1)((k+1)^2 + (k+1)^2)}{2}, && \text{because } (k+1)^2 \geq 2. \\
&= (k+1)^3, && \text{by algebra}
\end{aligned}
$$

Conclusion. Thus, since the base case and induction step hold, the P(n) is true for n ≥ 3.

Practice Final: 3. All The Machines!

Let $\Sigma = \{0, 1, 2\}$. Consider $L = \{w \in \Sigma^* :$ Every 1 in the string has at least one 0 before and after it$\}$.

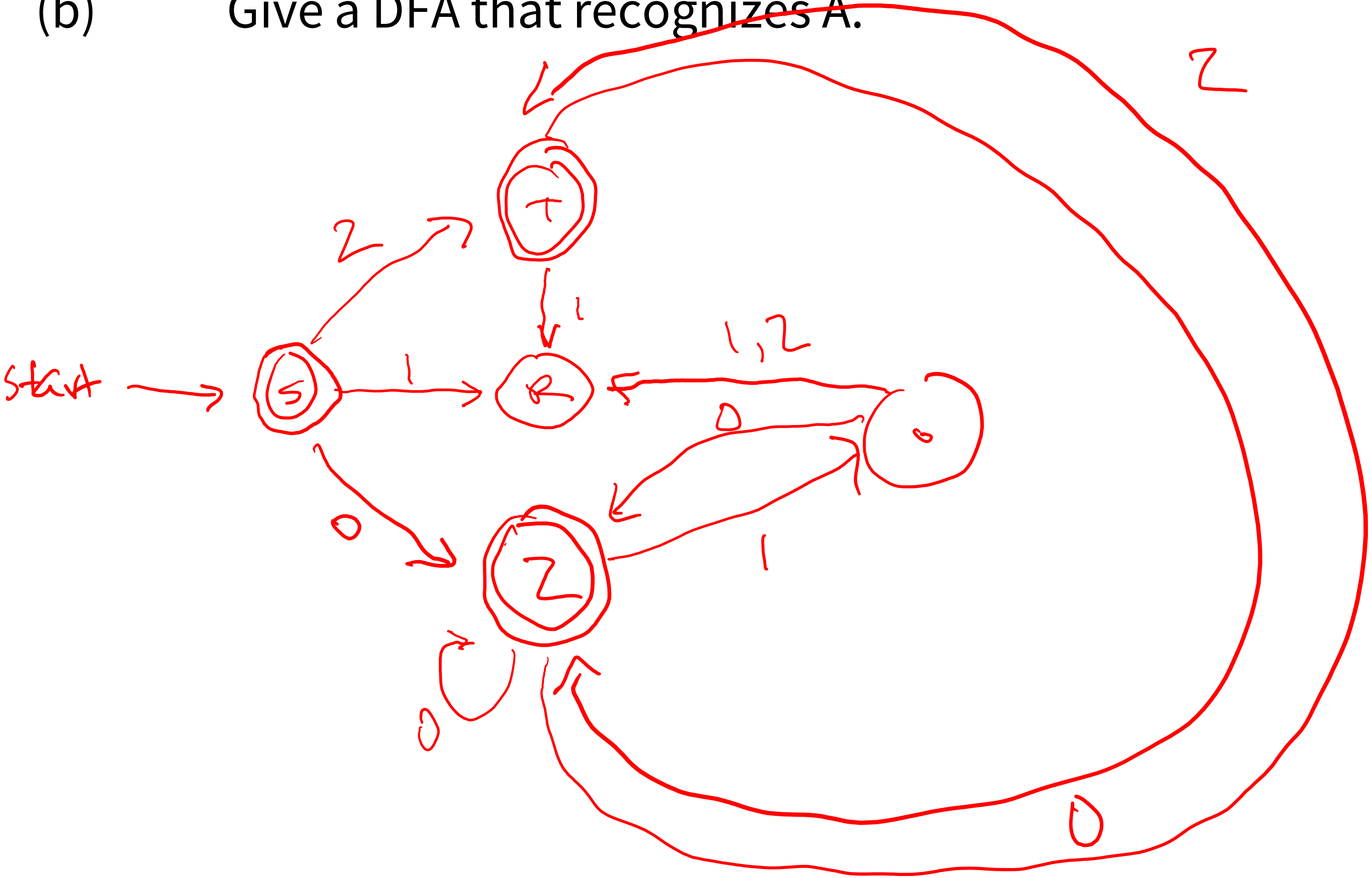
(a) Give a regular expression that represents A.

(b) Give a DFA that recognizes A.

(c) Give a CFG that generates A.

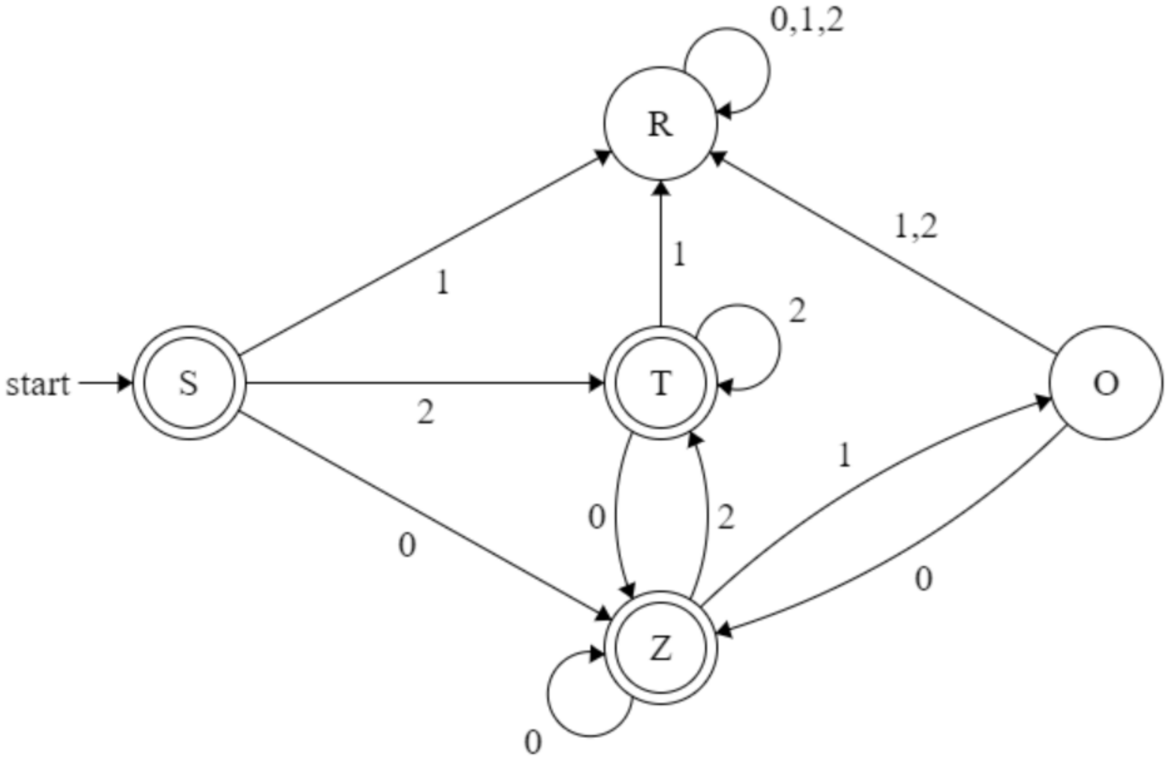
Practice Final: 3. All The Machines! Solution

(b) Give a DFA that recognizes A.



Practice Final: 3. All The Machines! Solution

(b) Give a DFA that recognizes A.



Practice Final: 3. All The Machines! Solution

(a) Give a regular expression that represents A.

$(0 \cup 2)^* \, (0(0 \cup 1 \cup 2)^*0)^* \, (0 \cup 2)^*$

(c) Give a CFG that generates A.

Practice Final: 3. All The Machines! Solution

(a) Give a regular expression that represents A.

$(0 \cup 2)^* \, (0(0 \cup 1 \cup 2)0)^* \, (0 \cup 2)^*$

(c) Give a CFG that generates A.

$\boldsymbol{S} \rightarrow 0\boldsymbol{S} \mid 2\boldsymbol{S} \mid \boldsymbol{T}$
$\boldsymbol{T} \rightarrow 0\boldsymbol{R}0\boldsymbol{T} \mid \boldsymbol{X}$
$\boldsymbol{R} \rightarrow 0 \mid 1 \mid 2$
$\boldsymbol{X} \rightarrow 0\boldsymbol{X} \mid 2\boldsymbol{X} \mid \varepsilon$

Practice Final: 4. Structural CFGs

CFG

Consider the following CFG: $\boldsymbol{S} \rightarrow \varepsilon \mid \boldsymbol{SS} \mid \boldsymbol{S}1 \mid \boldsymbol{S}01$. Another way of writing the recursive definition of this set, Q, is as follows:

- $\varepsilon \in Q$ Basis step
- If $S \in Q$, then $S1 \in Q$ and $S01 \in Q$
- If $S, T \in Q$, then $ST \in Q$.

} recursive step

Prove, by structural induction that if $w \in Q$, then w has at least as many 1's as 0's

Practice Final: 4. Structural CFGs Solution

Basis step: $\varepsilon \in Q$ ✓ ✓

recursive step: $S1 \in Q$, $S01 \in Q$ $\forall S \in Q$

$ST \in Q$ $\forall S, T \in Q$

$\#_0(w)$, $\#_1(w)$, $P(w) =$ "$\#_0(w) \leq \#_1(w)$"

Base case: $\#_0(\varepsilon) = 0$, $\#_1(\varepsilon) = 0$, $\#_0(\varepsilon) \leq \#_1(\varepsilon)$, $P(\varepsilon)$

IH: Assume $P(u)$, $P(v)$ for arbitrary $u, v \in Q$!

IS: Let u, v be arbitrary element in Q

case 1: By IH, $P(u)$ holds, $\#_0(u) \leq \#_1(u)$

$\#_0(u1) = \#_0(u)$, $\#_1(u1) = \#_1(u) + 1$

$\#_0(u1) = \#_0(u) \leq \#_1(u) \leq \#_1(u) + 1 = \#_1(u1)$

$P(u1)$

Practice Final: 4. Structural CFGs Solution

Case 2: $\#_0(u01) = \#_0(u)+1$, $\#_1(u01) = \#_1(u)+1$

By IH: $\#_0(u) \le \#_1(u)$

$P(u)$

$\#_0(u01) = \#_0(u)+1 \le \#_1(u)+1 = \#_1(u01)$

$\therefore P(u01)$

$P(u), P(v)$

Case 3: $\#_0(uv) = \#_0(u) + \#_0(v)$, $\#_1(uv) = \#_1(u) + \#_1(v)$

By IH: $\#_0(u) \le \#_1(u)$, $\#_0(v) \le \#_1(v)$

$\#_0(uv) = \#_0(u) + \#_0(v) \le \#_1(u) + \#_1(v)$

$\downarrow$

$P(uv)$

By priciple of structural induction, $P(w)\ \forall w \in Q$

Practice Final: BONUS Set Proof

A = {x : x ≡ k (mod 4)}, B = {x : x = 4r+k for some integer r}. Prove A = B for all integer k

Let k be an arbitrary integer

$A \subseteq B$

Let a be an arbitrary element in A

$a \equiv k \pmod 4 \rightarrow k \equiv a \pmod 4 \rightarrow 4 \mid a-k$

$4r = a-k$ for some integer r

$4r+k = a \rightarrow a = 4r+k$

$a \in B$

$4 \mid k-a$

$4r = k-a$

$4(-r) = -(k-a)$

$4(-r) = a-k$

Practice Final: BONUS Set Proof

A = {x : x ≡ k (mod 4)}, B = {x : x = 4r+k for some integer r}. Prove A = B for all integer k

$B \subseteq A$

Let b be an arbitrary element in B

$b = 4r + k$ for some integer r

$b - k = 4r \rightarrow 4r = b - k \rightarrow 4 \mid b - k$ as r is an integer

$k \equiv b \pmod 4 \rightarrow b \equiv k \pmod 4$

$b \in A$

$A \subseteq B$, $B \subseteq A$, $A = B$

Practice Final: 5. Tralse!

For each of the following answer True or False and give a short explanation of your answer.

(a) Any subset of a regular language is also regular.

(b) The set of programs that loop forever on at least one input is decidable.

(c) If $\mathbb{R} \subseteq A$ for some set A, then A is uncountable.

(d) If the domain of discourse is people, the logical statement
$\exists x\, (\mathrm{P}(x) \rightarrow \forall y\, (x \neq y \rightarrow \neg P(y))$
can be correctly translated as "There exists a unique person who has property P".

(e) $\exists x\, (\forall y\, \mathrm{P}(x, y)) \rightarrow \forall y\, (\exists x\, \mathrm{P}(x, y))$ is true regardless of what predicate P is.

Practice Final: 5. Tralse! Solution

(a) Any subset of a regular language is also regular.

$0^* 1^*$

Σ^*

$0^k 1^k$

Practice Final: 5. Tralse! Solution

(b) The set of programs that loop forever on at least one input is decidable.

False

Practice Final: 5. Tralse! Solution

(c) If $\mathbb{R} \subseteq A$ for some set A, then A is uncountable.

True

surjection $f: \mathbb{N} \to A$ Assume A is countable

$A - A/R$

i	$f(i)$
0	x_1
1	~~π e R~~
2	π s
3	⋮
⋮	⋮

Practice Final: 5. Tralse! Solution

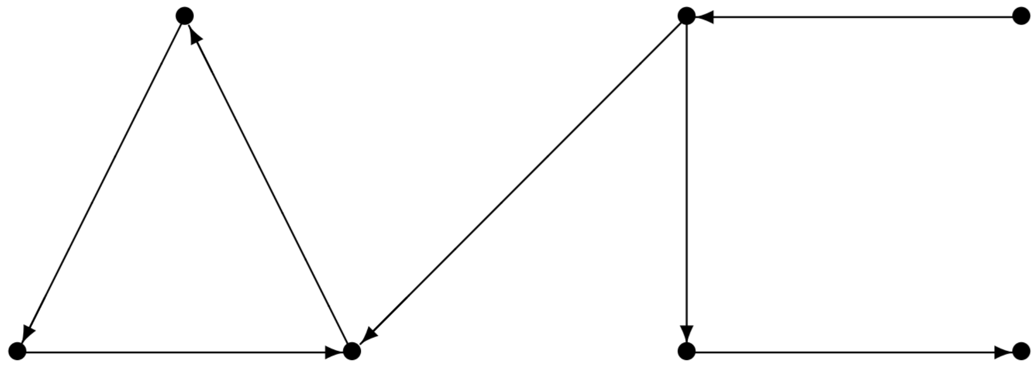
(d) If the domain of discourse is people, the logical statement
$\exists x\ (P(x) \rightarrow \forall y\ (x \neq y \rightarrow \neg P(y))$
can be correctly translated as “There exists a unique person who has property P”.

Practice Final: 5. Tralse! Solution

(e) $\exists x\, (\forall y\, P(x, y)) \rightarrow \forall y\, (\exists x\, P(x, y))$ is true regardless of what predicate P is.

Practice Final: 6. Regularly Irregular

The following is the graph of a binary relation *R*.

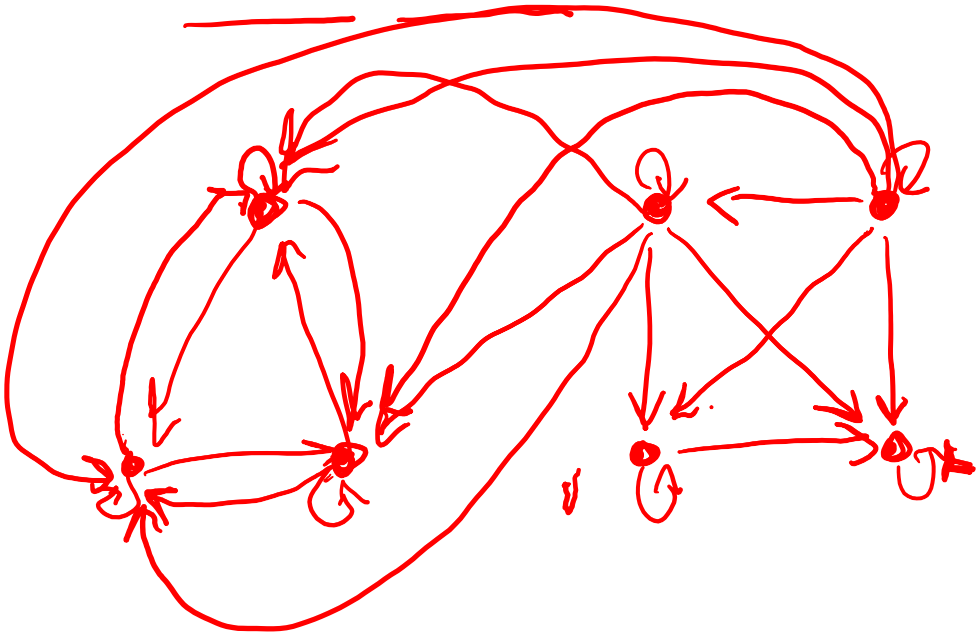


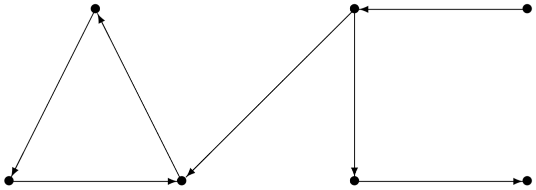
(a) Draw the transitive-reflexive closure of *R*.

(b) Let $S = \{(X, Y) : X, Y \in \mathcal{P}(\mathbb{N}) \wedge X \subseteq Y\}$.
Recall that R is antisymmetric iff $((a, b) \in R \wedge a \neq b) \rightarrow (b, a) \notin$ R.
Prove that *S* is antisymmetric.

Practice Final: 6. Regularly Irregular Solution

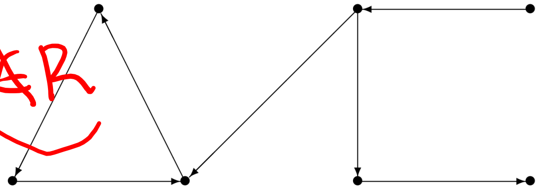
(a) Draw the transitive-reflexive closure of R.





Practice Final: 6. Regularly Irregular Solution

(b) Let $S = \{(X, Y) : X, Y \in \mathcal{P}(\mathbb{N}) \wedge X \subseteq Y\}$.
Recall that R is antisymmetric iff $((a, b) \in R \wedge a \neq b) \rightarrow (b, a) \notin R$
Prove that S is antisymmetric.

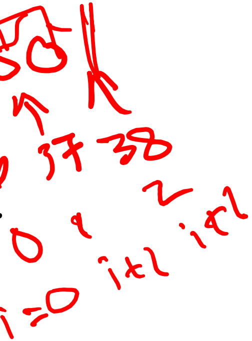


Suppose $(a,b) \in S$ and $a \neq b$. By def of S, $a \subset b$. By def of $\subset$, there is some $x \in b$ where $x \notin a$. Then, $b \not\subseteq a$, so $(b,a) \notin R$. Therefore, S is antisymmetric.

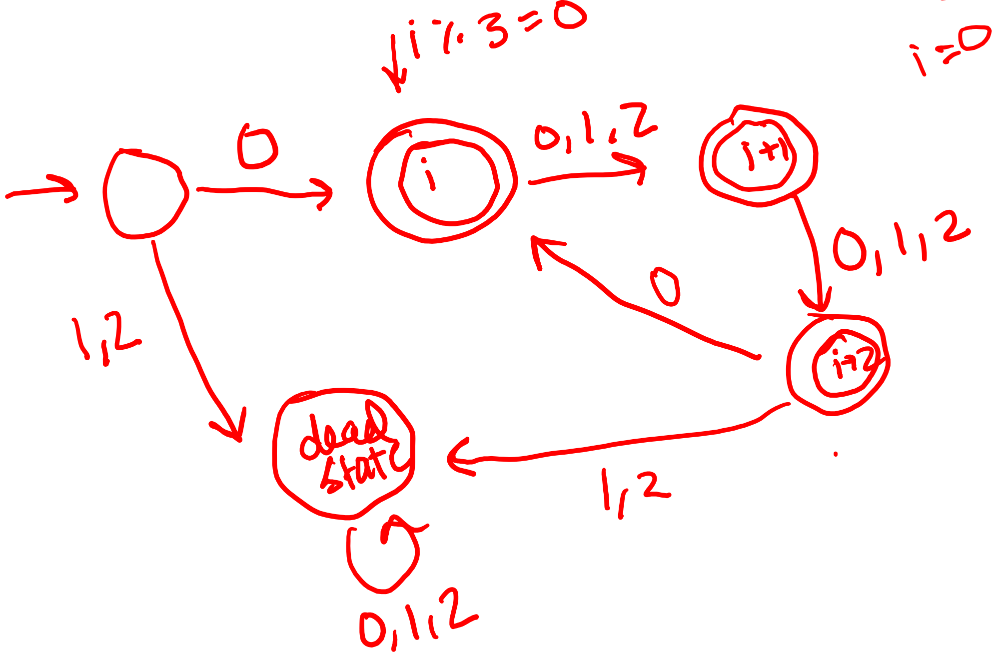
Practice Final: 8. Modern DFAs

Let Σ = {0, 1, 2}.

Construct a DFA that recognizes exactly strings with a 0 in all positions i where $i\%3 = 0$.



reject	accept
ε	0
1	01
2	012
0111	0110
	0000



Practice Final: 8. Modern DFAs Solution

Let $\Sigma = \{0, 1, 2\}$. Construct a DFA that recognizes exactly strings with a 0 in all positions i where $i \% 3 = 0$.

That’s All, Folks!

Any questions?