Review of Informatics 1 Computation & Logic

Basics for the exam

Truth Tables

P	Q	not(P)	P and Q	P or Q	P → Q	P ↔ Q
t	t	f	t	t	t	t
t	f	f	f	t	f	f
f	t	t	f	t	t	f
f	f	t	f	f	t	t

Inconsistencies

$$\underbrace{\overset{t}{P} \text{ and } \underbrace{\text{not}(\overset{t}{P})}_{f}}_{f}$$

$$\underbrace{\overset{f}{P} \text{ and } \underbrace{\text{not}(\overset{f}{P})}_{t}}_{f}$$

Contingencies

$$\underbrace{\underbrace{(\underset{t}{P} \text{ or } \underset{t}{Q})}_{t} \rightarrow \underset{t}{P}}_{t}$$

$$\underbrace{\underbrace{(\underset{t}{P} \text{ or } \underset{f}{Q})}_{t} \rightarrow \underset{t}{P}}_{t}$$

$$\underbrace{\underbrace{(\underset{f}{P} \text{ or } \underset{t}{Q})}_{t} \rightarrow \underset{f}{P}}_{f}$$

$$\underbrace{\underbrace{(\underset{f}{P} \text{ or } \underset{f}{Q})}_{f} \rightarrow \underset{f}{P}}_{t}$$

Tautologies

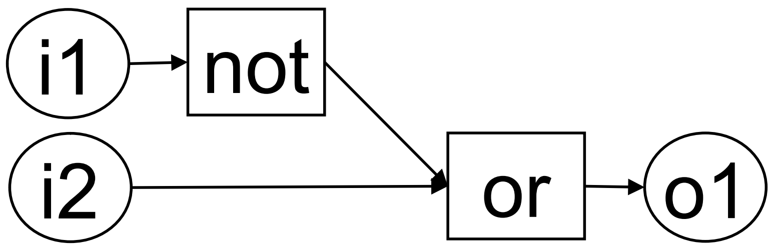
$$\underbrace{\underbrace{\underset{t}{P} \text{ and } (\underbrace{\underset{t}{P} \to \underset{t}{Q}}_{t})}_{t} \to \underset{t}{Q}}_{t}$$

$$\underbrace{\underbrace{\underset{t}{P} \text{ and } (\underbrace{\underset{t}{P} \to \underset{f}{Q}}_{f})}_{f} \to \underset{f}{Q}}_{t}$$

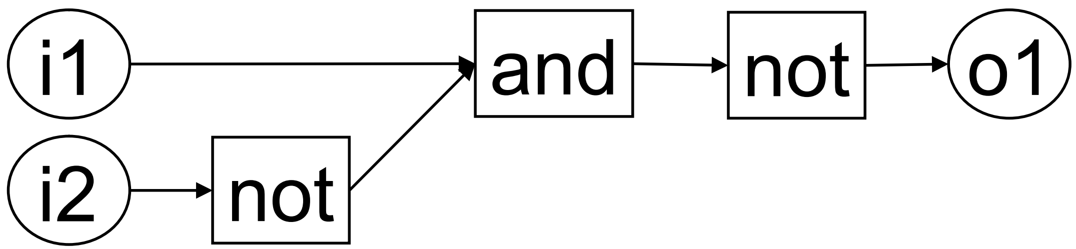
$$\underbrace{\underbrace{\underset{f}{P} \text{ and } (\underbrace{\underset{f}{P} \to \underset{t}{Q}}_{t})}_{t} \to \underset{t}{Q}}_{t}$$

$$\underbrace{\underbrace{\underset{f}{P} \text{ and } (\underbrace{\underset{f}{P} \to \underset{f}{Q}}_{t})}_{f} \to \underset{f}{Q}}_{t}$$

Typical Problem



equivalent ?



not(i1) or i2

equivalent ?

not(I1 and not(i2))

Proof Rules

Proof	Sub-proofs		
$F \vdash A$	$A \in F$		
$F \vdash A$ and B	$F \vdash A$	$F \vdash B$	
$F \vdash A$ or B	$F \vdash A$		
$F \vdash A$ or B	$F \vdash B$		
$F \vdash C$	A or $B \in F$	$[A \mid F] \vdash C$	$[B \mid F] \vdash C$
$F \vdash B$	$A \rightarrow B \in F$	$F \vdash A$	
$F \vdash A \rightarrow B$	$[A \mid F] \vdash B$		

Proofs (and Proof Trees)

- $[B \rightarrow C] \vdash (A \text{ or } B) \rightarrow (A \text{ or } C)$
  - $[(A \text{ or } B), B \rightarrow C] \vdash (A \text{ or } C)$
    - $(A \text{ or } B) \in [(A \text{ or } B), B \rightarrow C]$
    - $[A, (A \text{ or } B), B \rightarrow C] \vdash (A \text{ or } C)$
      - $[A, (A \text{ or } B), B \rightarrow C] \vdash A$
        - $A \in [A, (A \text{ or } B), B \rightarrow C]$
    - $[B, (A \text{ or } B), B \rightarrow C] \vdash (A \text{ or } C)$
      - $[B, (A \text{ or } B), B \rightarrow C] \vdash C$
        - $B \rightarrow C \in [B, (A \text{ or } B), B \rightarrow C]$
        - $[B, (A \text{ or } B), B \rightarrow C] \vdash B$
          - $B \in [B, (A \text{ or } B), B \rightarrow C]$

Proof Rules for Negation

Proof	Sub-proofs	
$F \vdash A$	$F \vdash \text{false}$	
$F \vdash \text{not}(A)$	$[A \mid F] \vdash \text{false}$	
$F \vdash B$	$\text{not}(A) \in F$	$F \vdash A$
$F \vdash A$	$F \vdash \text{not}(\text{not}(A))$	

The purpose of these rules is to provide positive evidence that an expression is false.

Negation as Failure

Replace <u>all</u> the previous negation rules with:

Proof	Sub-proofs
$F \vdash \text{not}(A)$	$F \nvdash A$

Where $F \nvdash A$ means a proof can't be found for A from F

This makes the closed world assumption that F contains all the axioms pertinent to the problem and that the proof search is complete.

Equivalences

Basic equivalences to remember are:

not(not(A)	is equivalent to	A
$A \rightarrow B$	is equivalent to	not(A) or B
$A \leftrightarrow B$	is equivalent to	$(A \rightarrow B)$ and $(B \rightarrow A)$
not(A or B)	is equivalent to	not(A) and not(B)
not(A and B)	is equivalent to	not(A) or not(B)
A or (B and C)	is equivalent to	(A or B) and (A or C)
A and (B or C)	is equivalent to	(A and B) or (A and C)

Conversion to Clausal Form

(a and not(b) → c) and a and not(c)

P → Q equivalent to not(P) or Q

(not(a and not(b)) or c) and a and not(c)

not(P and Q) equivalent to not(P) or not(Q)

(not(a) or not(not(b)) or c) and a and not(c)

not(not(P)) equivalent to P

(not(a) or b or c) and a and not(c)

(P or …) and … to [[P,…],…]

[[not(a),b,c], [a], [not(c)]]

A Resolution Proof

To prove b from [[not(a),b,c], [a], [not(c)]]

show that [[not(b)], [not(a),b,c], [a], [not(c)]]
is inconsistent

[not(b)]　　[not(a),b,c]

[not(a), c]　　[a]

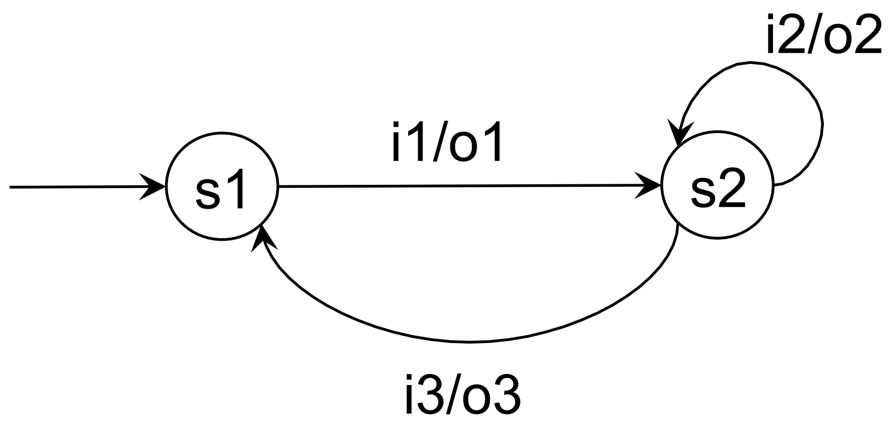
[c]　　[not(c)]

[ ]

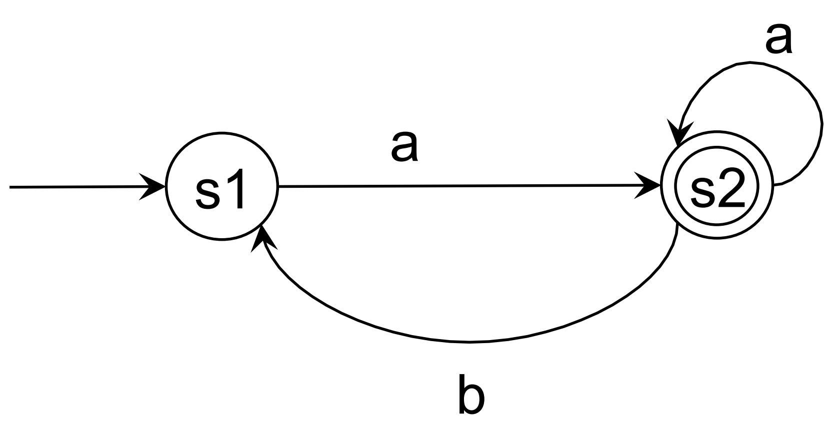
Temporal Proof Rules

Proof	Sub-proofs
$(S,J) \vdash A$	access(J, S, F) $\quad F \vdash A$
$(S,J) \vdash A$ and B	$(S,J) \vdash A$ $\quad (S,J) \vdash B$
$(S,J) \vdash A$ or B	$(S,J) \vdash A$
$(S,J) \vdash A$ or B	$(S,J) \vdash B$
$(S,J) \vdash$ not(A)	$(S,J) \nvdash A$
$(S,J) \vdash$ next(A)	$(S,J+1) \vdash A$
$(S,J) \vdash$ prev(A)	$(S,J-1) \vdash A$
$(S,J) \vdash$ e_future(A)	$(S,K) \vdash A$ for some $K > J$
$(S,J) \vdash$ e_past(A)	$(S,K) \vdash B$ for some $K < J$
$(S,J) \vdash$ a_future(A)	$(S,K) \vdash A$ for all $K > J$
$(S,J) \vdash$ a_past(A)	$(S,K) \vdash A$ for all $K < J$

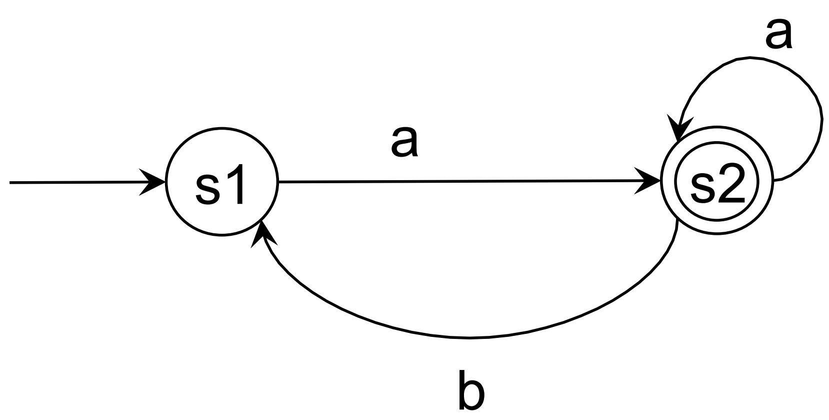
Transducer FSMs



Acceptor FSMs

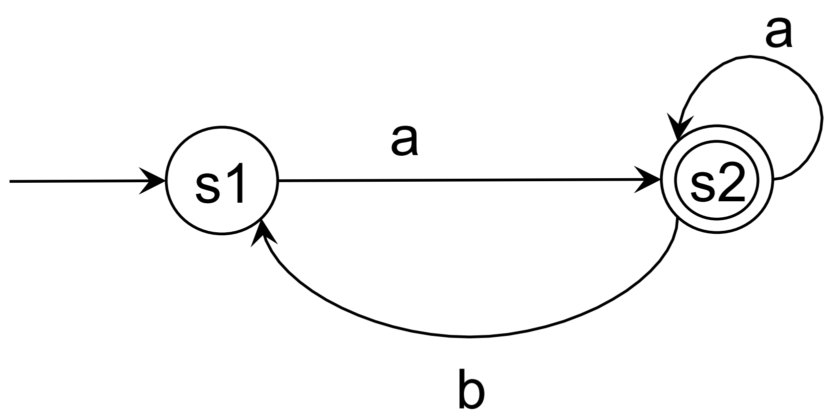


Traces



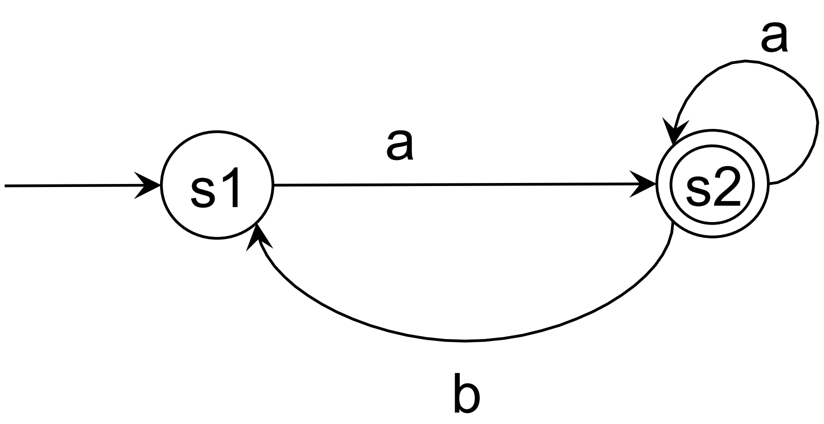
s1, a, s2, a, s2, b, s1, a, s2

Transition Function

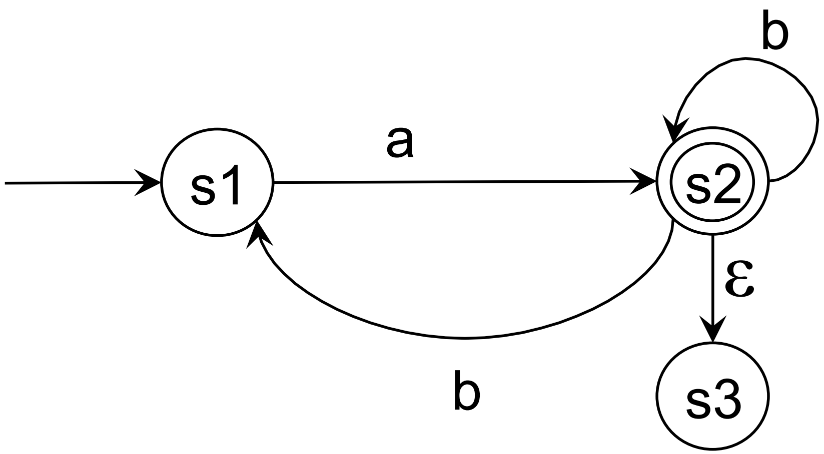


	a	b
s1	s2	-
s2	s2	s1

Deterministic v Nondeterministic

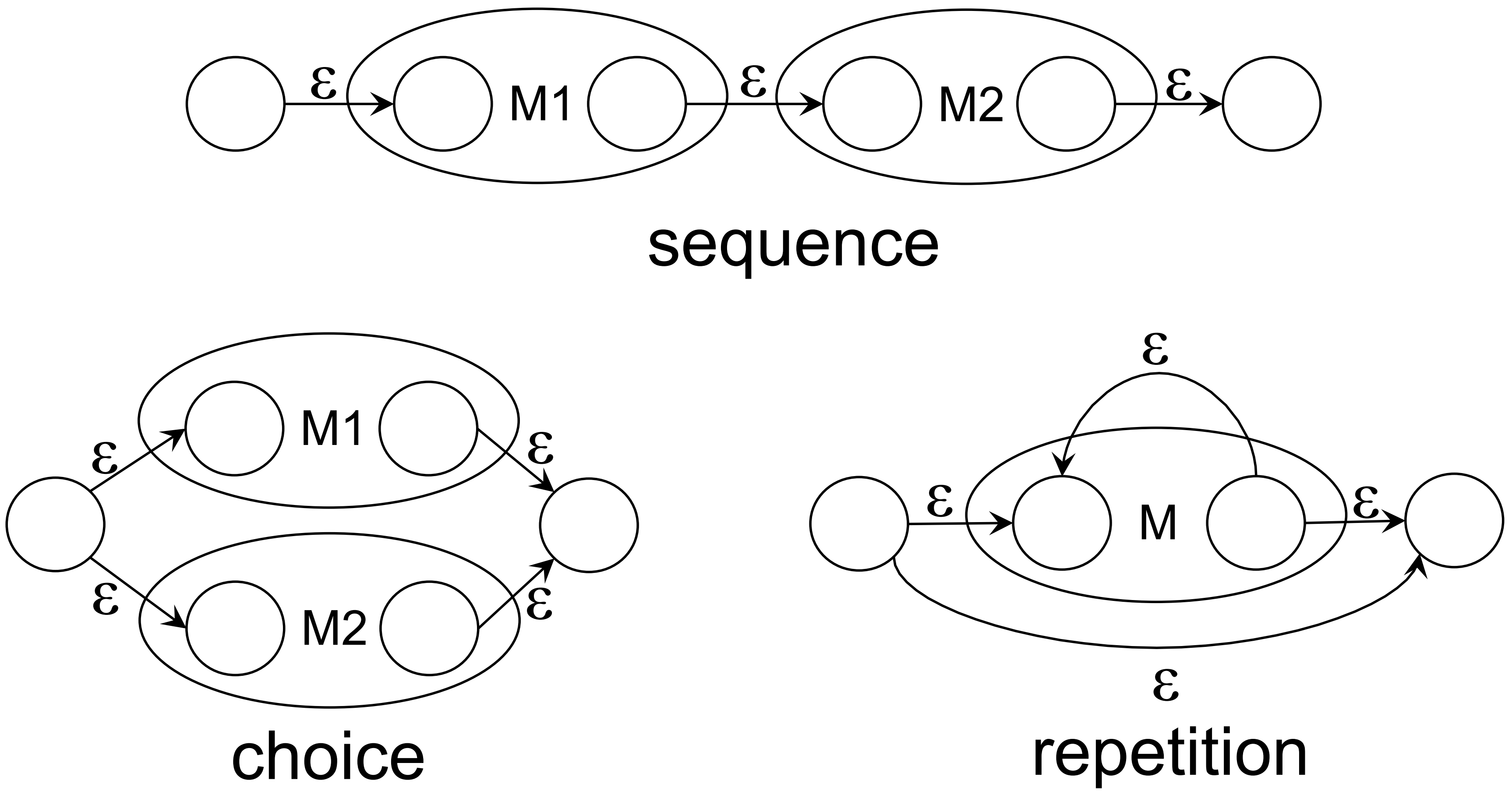


Deterministic



Nondeterministic

Structured FSM Design



Regular Expressions

E_1E_2	Sequence
$E_1\|E_2$	Choice
E^*	Repetition

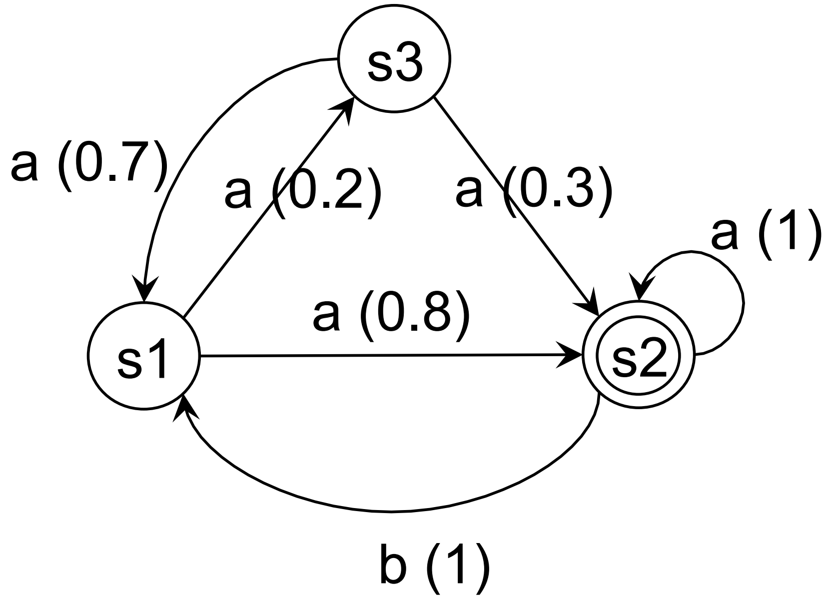
Half of Kleene's Theorem

For every regular expression we can build a FSM to accept the language defined by it.

Limits of FSMs and Regular Expressions

Some languages can't be defined by FSMs or regular expressions - languages that require us to count up to arbitrarily high numbers for example.

Probabilistic FSMs



Thanks

and good luck with the exam