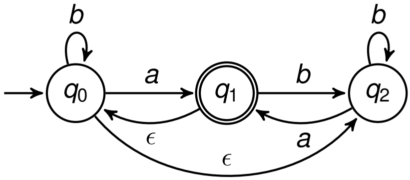
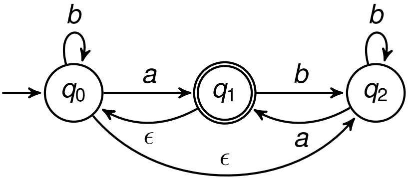
CS 228 : Logic in Computer Science

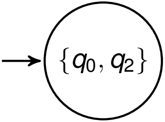
Krishna. S

ϵ-NFA

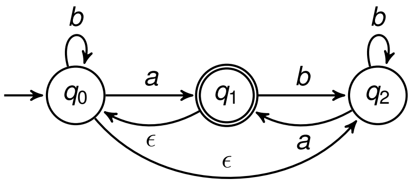


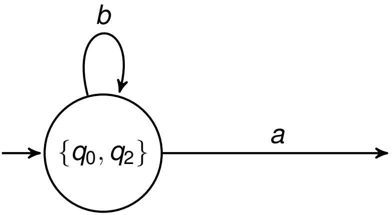
ϵ-NFA





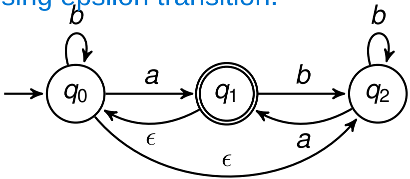
ϵ-NFA

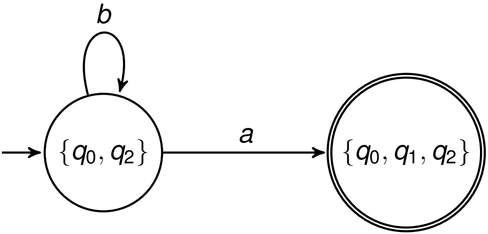




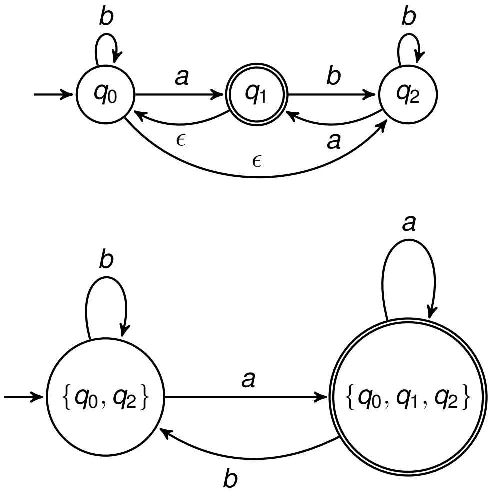
ϵ-NFA

epsilon closure of a state is the set of all state reachable from the given state using epsilon transition.





ϵ-NFA

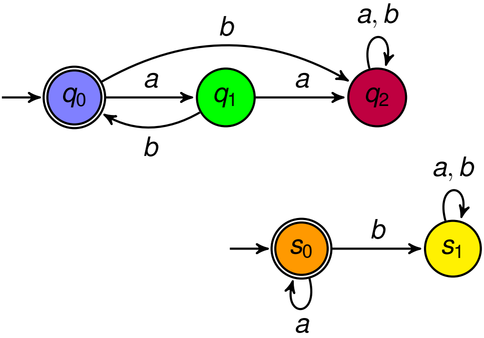


ϵ-NFA and DFA

- ϵ-close the initial states of the ϵ-NFA to obtain initial state of DFA
- From a state S, compute $\Delta(S, a)$ and ϵ-close it
- All states in the DFA are ϵ-closed
- Final states are those which contain a final state of the ϵ-NFA

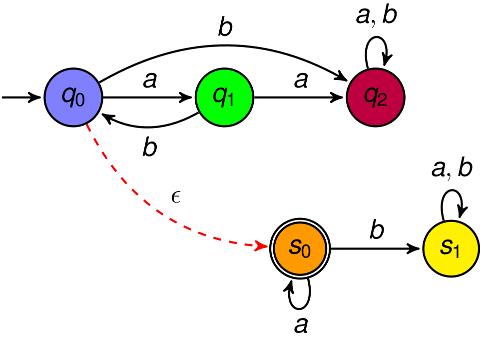
Closure under Concatenation

- Given regular languages L_1, L_2, is $L_1.L_2$ regular



Closure under Concatenation

- Given regular languages L_1, L_2, is $L_1.L_2$ regular?



Formulae to Automaton

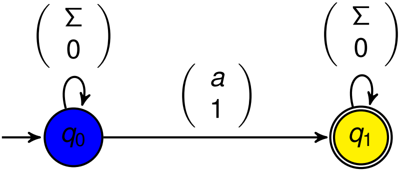
- FO-definable $\Rightarrow$ regular
- Given an FO formula φ, construct a DFA A_φ such that $L(\varphi) = L(A_\varphi)$
- If $L(A_\varphi) = \emptyset$, then φ is unsatisfiable
- If $L(A_\varphi) \neq \emptyset$, then φ is satisfiable

FO to Regular Languages

- Every FO sentence φ over words can be converted into a DFA A_φ such that $L(\varphi) = L(A_\varphi)$.
- Start with atomic formulae, construct DFA for each of them.
- Conjunctions, disjunctions, negation of formulae easily handled via union, intersection and complementation of of respective DFA
- Handling quantifiers?

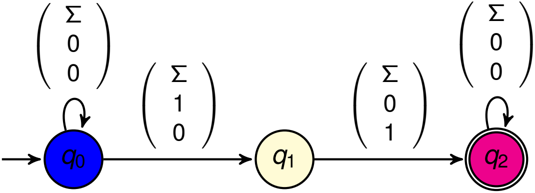
Atomic Formulae to DFA

- $Q_a(x)$: All words which have an a. Need to fix a position for x, where a holds.
- *baab* satisfies $Q_a(x)$ with assignment $x = 1$ or $x = 2$.
- Think of this as $\begin{matrix} baab \\ 0010 \end{matrix}$ or $\begin{matrix} baab \\ 0100 \end{matrix}$
- The first row is over Σ, and the second row captures a possible assignment to x
- Think of an extended alphabet $\Sigma' = \Sigma \times \{0, 1\}$, and construct an automaton over Σ'.
- Deterministic, not complete.



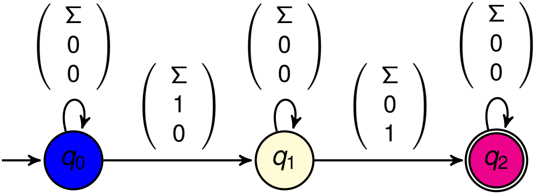
Atomic Formulae to DFA : $S(x, y)$

- *bab* satisfies $S(x, y)$ with assignment $x = 0$ or $y = 1$ or $x = 1, y = 2$.
- Think of this as $\begin{array}{c} bab \\ 100 \\ 010 \end{array}$ or $\begin{array}{c} bab \\ 010 \\ 001 \end{array}$
- The first row is over Σ, and the second, third rows capture a possible assignment to x, y
- Think of an extended alphabet $\Sigma' = \Sigma \times \{0, 1\}^2$, and construct an automaton over Σ'.
- Deterministic, not complete.



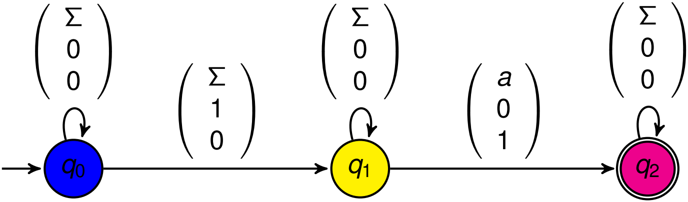
Atomic Formulae to DFA : $x < y$

- *bab* satisfies $x < y$ with assignment $x = 0$ or $y = 1$ or $x = 1, y = 2$ or $x = 0, y = 2$.



Simple Formulae to DFA

- $x < y \wedge Q_a(y)$
- $\Sigma' = \Sigma \times \{0, 1\} \times \{0, 1\}$
- Obtain intersection of DFA for $x < y$ and $Q_a(y)$

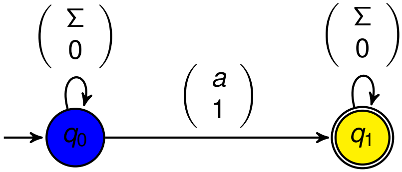


Formulae to DFA

- Given $\varphi(x_1, \ldots, x_n)$, a FO formula over Σ, consider the extended alphabet
$$\Sigma' = \Sigma \times \{0, 1\}^n$$
- Assign values to x_i at every position as seen in the cases of atomic formulae
- Keep in mind that every x_i can be assigned 1 at a unique position

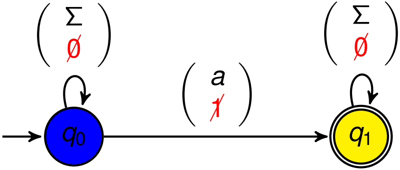
Quantifiers

- $Q_a(x)$



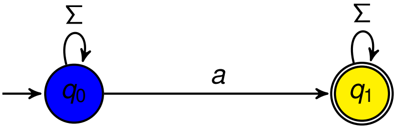
Handling Quantifiers

- $\exists x Q_a(x)$



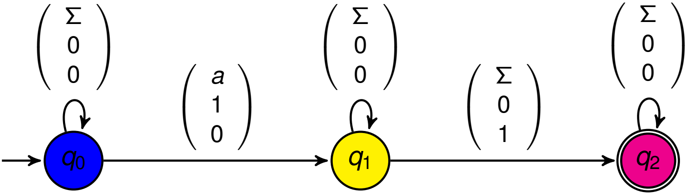
Handling Quantifiers

- $\exists x Q_a(x)$



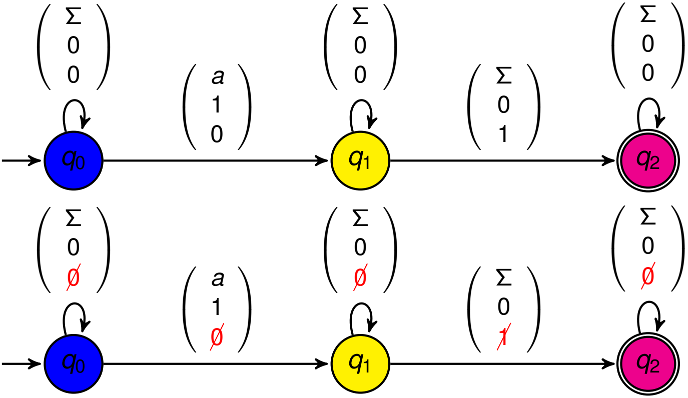
Handling Quantifiers

- $Q_a(x) \wedge \exists y(x < y)$



Handling Quantifiers

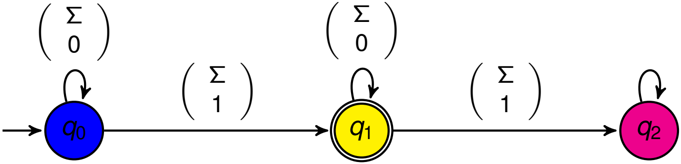
- $Q_a(x) \wedge \exists y(x < y)$



Handling Quantifiers: $\forall x(x \neq x)$

Handling Quantifiers: $\forall x(x \neq x)$

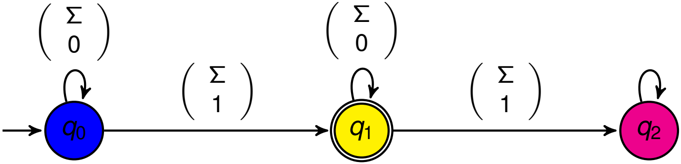
- $(x = x)$



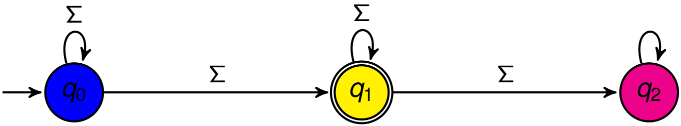
- $\exists x(x = x)$

Handling Quantifiers: $\forall x(x \neq x)$

- $(x = x)$



- $\exists x(x = x)$



- $\neg\exists x(x = x)$

Handling Quantifiers : Summary

- Let $L \subseteq (\Sigma \times \{0,1\}^n)^*$ be defined by $\varphi(x_1, \ldots, x_n)$.
- Let $f : (\Sigma \times \{0,1\}^n)^* \to (\Sigma \times \{0,1\}^{n-1})^*$ be the projection $f(w, c_1, \ldots, c_n) = (w, c_1, \ldots, c_{n-1})$.
- Then $\exists x_n \varphi(x_1, \ldots, x_{n-1})$ defines $f(L)$.

Handling Quantifiers : Done on Board

- $\exists x \forall y [x > y \vee \neg Q_a(x)] = \exists x [\neg \exists y [x \leqslant y \wedge Q_a(x)]]$
- Draw the automaton for $[x \leqslant y \wedge Q_a(x)]$
- Project out the y-row
- Determinize it, and complement it
- Fix the x-row : Intersect with $\begin{pmatrix} \Sigma \\ 0 \end{pmatrix}^* \begin{pmatrix} \Sigma \\ 1 \end{pmatrix} \begin{pmatrix} \Sigma \\ 0 \end{pmatrix}^*$
- Project the x-row