Automaty a formální jazyky

Přednáška III.

Ekvivalence a minimalizace automatů

Konečné automaty - nedeterministické

Hledání dosažitelných stavů - algoritmus

- Princip: K již známým dosažitelným stavům přidáváme postupně v jednotlivých krocích ty stavy, do nichž se lze dostat jedním symbolem z již známých dosažitelných stavů.
- Počátek: počáteční stav je vždy dosažitelný.
- Konec: v daném kroku už nepřibude žádný další stav.

Redukovaný automat

- **Definice:** Konečný automat je **redukovaný**, když jsou všechny jeho stavy dosažitelné a nejsou si ekvivalentní.
- **Definice:** Automat A je **reduktem** automatu B, jestliže jsou si automaty ekvivalentní a A je redukovaný automat.
- **Věta**: Ke každému konečnému automatu existuje jeho redukt.

S pomocí redukce rozhodujeme

- **Definuje automat vůbec nějaký jazyk?** Jen tehdy, má-li dosažitelný alespoň jeden koncový stav.
- **Definuje automat celý uzávěr nad abecedou?** Pokud ano, potom jej lze redukovat na jeden jediný stav.
- **Jsou dva automaty ekvivalentní?** Pokud ano, jdou oba redukovat na stejný automat.

Poznámka pod čarou: slovem automat vždy myslíme námi definovaný konečný deterministický automat.

NKA I.:
Nedeterministický konečný automat

- „Může být najednou ve více stavech.“
- Čím se liší od DKA?
  - Může mít méně stavů - snazší návrh i realizace
  - Vhodnější pro určité aplikace (vyhledávání)
  - Rozdíl je v přechodové funkci

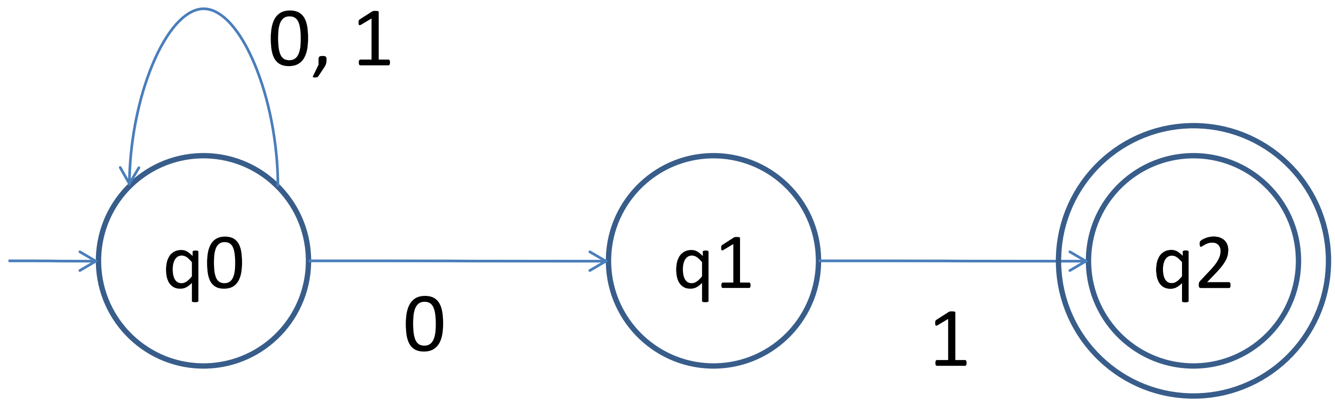
NKA II.: Definice

- **Nedeterministický konečný automat** je každá uspořádaná pětice

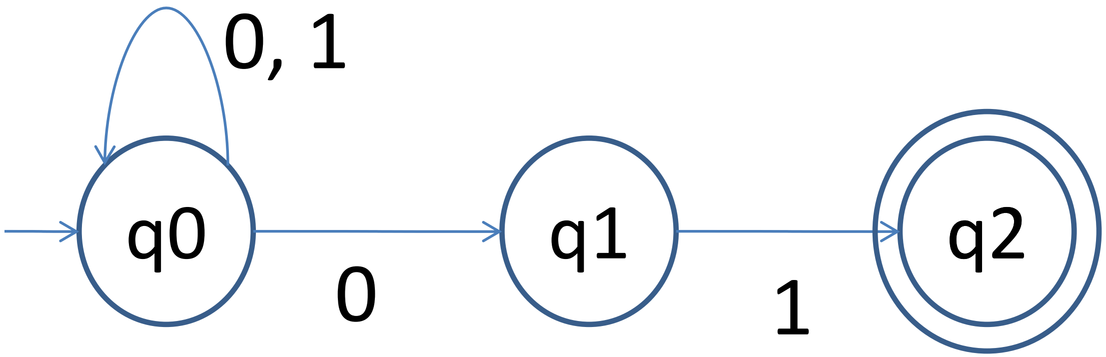
  $A = \{Q, \sum, \delta, q_0, F\}$ kde:
  - Q je konečná neprázdná množina (stavy)
  - $\sum$ je konečná neprázdná množina (vstupní abeceda)
  - δ je přechodová funkce, která ke každé uspořádané dvojici ($q \in Q, a \in \sum$) přiřadí podmnožinu Q
  - $q_0 \in Q$ (počáteční stav)
  - $F \subseteq Q$ je konečná množina (rozpoznávací stavy).

NKA III. - příklad

- Automat, který rozpoznává řetězce nul a jedniček, jež končí sekvencí 01. $\sum = \{0,1\}$

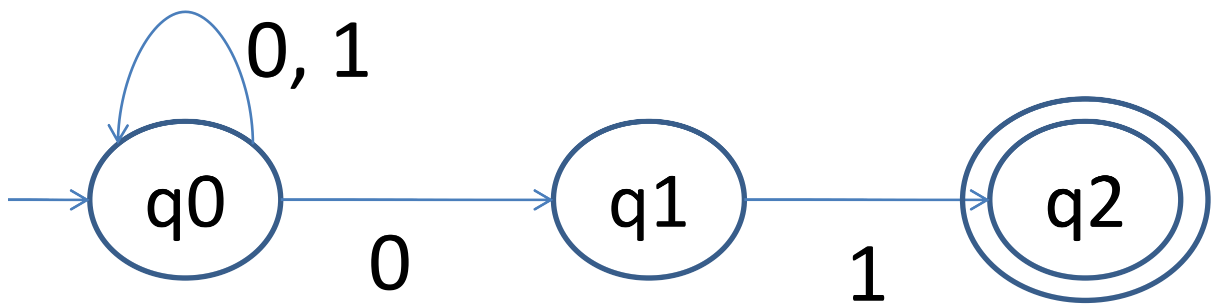


NKA III. – srovnání s DKA



- Počet vstupů a výstupů se u jednotlivých stavů nerovná.
- Výpočet někdy „zemře“ – není-li pro daný vstup definovaný přechod, stav se v dalším kroce „ruší“.

NKA III. –definice δ tabulkou



	0	**1**
q0 (poč.)	{q0, q1}	{q0}
q1	{}	{q2}
q2	{}	{}

Rozšířená přechodová funkce

- Analogicky k DKA:
  - Přechodová funkce δ zavedená v definici má jako vstup **stav** a **právě jeden symbol** vstupní abecedy.
  - Rozšířená přechodová funkce δ* má jako vstup **stav** a **slovo** sestavené nad vstupní abecedou. Výstupem je opět podmnožina Q.

Jazyk a NKA

- Neformálně: *Automat rozpoznává slovo, pokud po ukončení vstupu je alespoň v jednom rozpoznávacím stavu (výpočet „nezemře").*
- Formálně: Je-li $A = \{Q,\sum,\delta,q_0,F\}$ nedeterministický konečný automat, potom jazyk L je přijímán automatem právě tehdy když:

  $L = \{w; \delta^*(q0,w) \cap F \neq \emptyset\}$

Souvislost mezi NKA a DKA

- **Věta**: Každý jazyk, který lze popsat s pomocí NKA, lze popsat také s pomocí DKA.
- Důsledek: NKA lze převést na ekvivalentní DKA.
- V praxi mohou mít i podobný počet stavů, ale DKA má (většinou) více přechodů.
- Při převodu: kde má NKA *n* stavů, má vygenerovaný DKA 2^n stavů.

Převod NKA na DKA I.

Podmnožinová konstrukce

Máme automat $N = \{Q_N, \sum, \delta_N, q_0, F_N\}$

Hledáme automat $D = \{Q_D, \sum, \delta_D, q_0, F_D\}$

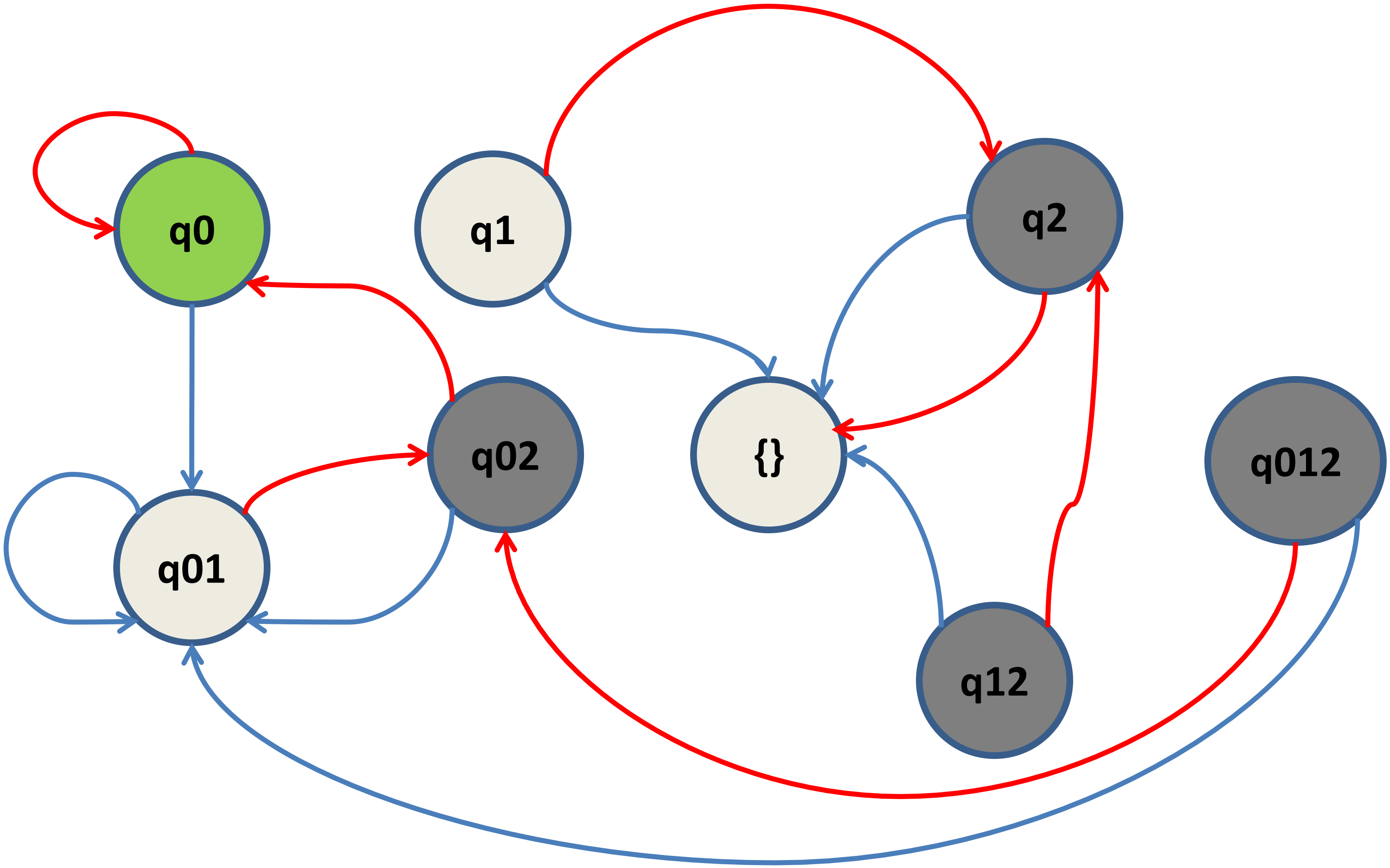
1. Q_D je množina všech podmnožin Q_N
2. F_D je množina podmnožin Q_N, takových, že obsahují alespoň jeden koncový stav automatu N.
3. Pro každou množinu S z podmnožin Q_N a pro každý vstupní symbol definujeme δ_D

   $\delta_D(S,a) = \cup_{p\ z\ S}\, \delta_N(p,a)$

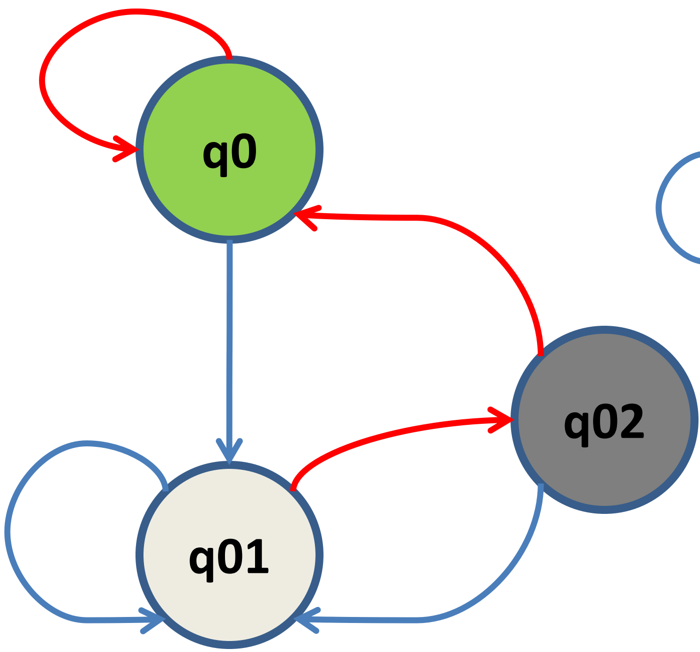
Převod NKA na DKA II.

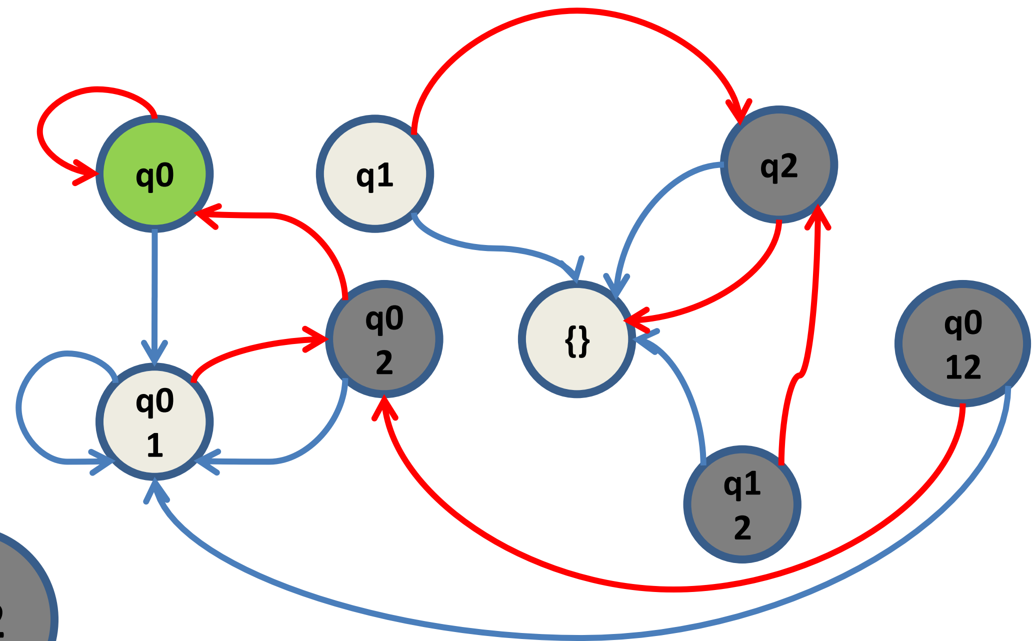
	0	1
{}	{}	{}
{q0}	{q0, q1}	{q0}
{q1}	{}	{q2}
{q2}	{}	{}
{q0, q1}	{q0, q1}	{q0,q2}
{q0, q2}	{q0, q1}	{q0}
{q1, q2}	{}	{q2}
{q0,q1,q2}	{q0, q1}	{q0,q2}

Převod NKA na DKA III.



Převod NKA na DKA IV.





Užití NKA

- Vyhledávání: automat, který rozpozná určitá klíčová slova
- Příklad: NKA, který rozpozná slova **mes** a **myš:**