1. príklad

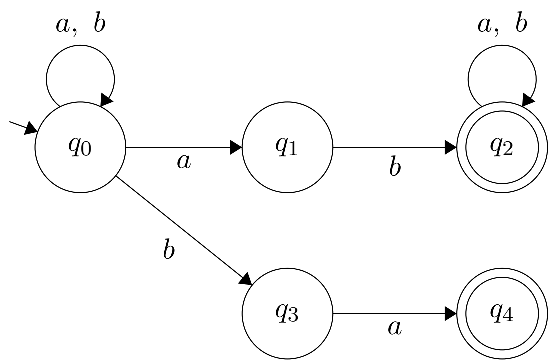
Zopakujme si tvorbu nedterministických automatov. Zostrojte NKA pre jazyky:

(a) $L_1 = \{xba \mid x \in \{a,b\}^*\} = \{a,b\}^*\{ba\}$

(b) $L_2 = \{xaby \mid x, y \in \{a,b\}^*\} = \{a,b\}^*\{ab\}\{a,b\}^*$

(c) $L_3 = L_1 \cup L_2 = \{w = xba \vee zaby \mid x, y, z \in \{a,b\}^*\} = \{a,b\}^*\{ba\} \cup \{a,b\}^*\{ab\}\{a,b\}^*$

L_1 a L_2 nechávame na precvičenie a zostrojme rovno NKA pre jazyk L_3.



Z obrázku je zrejmé, že v stave q_0 sa prečíta prefix x resp. z. Prechodom cez q_1 až do q_2 prečítame podslovo ab a v stave q_2 sa dočíta sufix y. Stav q_2 tak akceptuje slová z jazyka L_2. Prechodom cez q_3 do q_4 prečítame podslovo (sufix) ba, čiže stav q_4 akceptuje slová z jazyka L_1.

Skúste tento automat previesť podmnožinovou konštrukciou na DKA. Koľko stavov bude mať? Viete pre tieto jazyky vytvoriť aj DKA priamo? Majú viac alebo menej stavov ako automat vytvorený z NKA?

2. príklad

Nech A je konečný automat rozpoznávajúci jazyk $L - \{\lambda\}$; $L = L(A)$.

(a) Navrhnite a zdôvodnite konštrukciu, ktorá k danému nedeterministickému automatu A vytvorí nedeterministický automat A', taký, že $L(A) = L(A')$ s nasledujúcimi vlastnosťami:

  (i) má počiatočný stav ${q_0}'$

  (ii) a jediný akceptačný stav ${q_F}'$

(iii) neexistuje prechod do $q_0{}'$

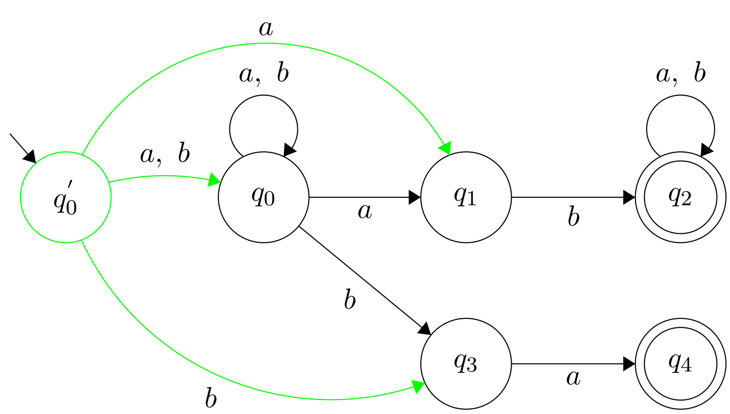
(iv) neexistuje prechod z $q_F{}'$

(b) Využite konštrukciu z bodu a) na zostrojenie KA pre jazyky $L^2,\ L^+,\ L^*$ $L_1L_2,\ L_1 \cup L_2$.

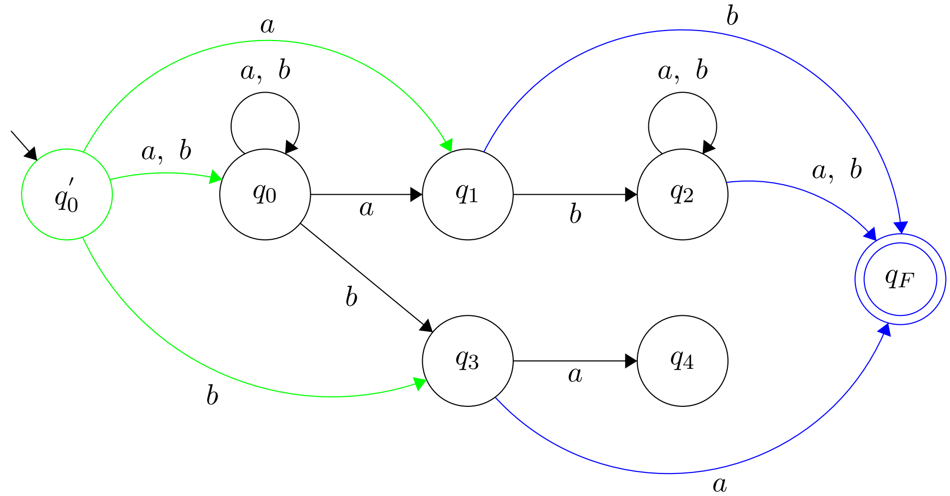
Riešenie 2. príkladu

Poďme sa spolu zamyslieť, čo znamenajú jednotlivé pravidlá. Pravidlo (i) samé o sebe nehovorí nič, v automate vždy máme jeden vstupný stav, ale v kombinácii s (iii) vidíme, že budeme musieť vyriešiť to, ak sa vraciame do vstupného stavu. Takže budeme musieť vytvoriť nový vstupný stav, z ktorého budú prechody iba vychádzať, ale nevrátia sa do neho. Ako to budeme robiť? Kým sa dostaneme k všeobecnému riešeniu, bude jednoduchšie, ak budeme skúšať na príklade. Využime automat pre L_3, ktorý sme zostrojili v prechádzajúcej časti. A poďme si na ňom ukázať, čo bude znamenať pridanie nového vstupného stavu.

Pridáme stav q_0'. Vieme, že z neho budú prechody iba vychádzať. Ale ako? Pozrime na pôvodný vstupný stav q_0, z neho existujú tri prechody. Jeden po a do q_1, druhý po b do q_3, a tretí – slučka po a aj b do q_0. Tieto prechody by sme mali zachovať. Takže z nášho nového vstupného stavu q_0' pôjdu prechody tak isto. V obrázku znázornené zelenou. Prečo sme zachovali všetky prechody, nestačil iba prechod do q_0? Nie, pozrite na slová ab a ba – obe sú v pôvodnom automate akceptované, keby sme spravili iba prechod do q_0, neakceptovali by sme ani jedno z týchto slov. Prečo je dôležitý prechod do q_0? Povedali sme si, že tam sa vytvárajú prefixy slova x resp. z, bez prechodu do q_0 by sme o ne prišli. Takto sme zachovali všetky možnosti a všetky prechody z nášho nového stavu iba vychádzajú, žiaden doňho nevchádza.



Čo s pravidlami (ii) a (iv)? Potrebujeme len jeden akceptačný stav, do ktorého prechody iba vchádzajú. Vytvorme teda stav q_F, ktorý bude akceptačný. Ako sa do neho dostaneme? Tak isto, ako sme sa dostali do q_2 alebo q_4. Čiže – z q_1 po b, z q_2 po a aj b, a z q_3 po a. V obrázku znázornené modrou.



Pôvodné akceptačné stavy už nie sú akceptačné. Boli všetky modré prechody potrebné? Ak by sme vynechali prechod z q_1 po b, neakceptovali by sme slovo ab. Ak by sme vynechali prechod z q_2, tak by sme zahodili sufixy y. V prípade stavu q_4 do neho išiel iba jeden prechod z q_3, ten sme zduplikovali. Zamyslite sa, či sa teraz nejaký výpočet skončí v stave q_4.

Dôležité je uvedomiť si, že my sme prechody iba pridávali. Tie pôvodné, ktoré nám slúžili ako predloha, sme nerušili, v automate zostali. Keďže sa jedná o nedeterministický automat, je v poriadku, ak z jedného stavu ide viac prechodov po rovnakom symbole abecedy.

Spravili sme si konštrukciu pre konkrétny automat. Ako by sme to zovšeobecnili? Potrebujeme vytvoriť postup, ktorý sa bude dať aplikovať na akýkoľvek NKA. Tak skúsme.

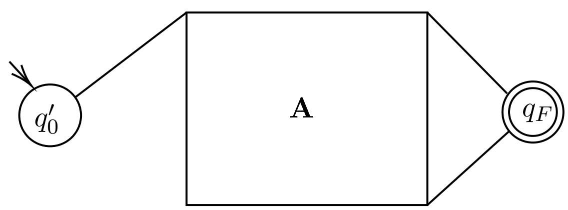
- Vezmeme pôvodný NKA, prechody aj stavy zostanú, zrušíme iba označenia vstupného a akceptačných stavov.

- Vytvoríme nový vstupný stav q_0'. Prechody z neho zostrojíme podľa prechodov z pôvodného vstupného stavu. Tj. pre všetky $\alpha \in \Sigma$ a všetky $p \in Q$ kde $\delta(q_0, \alpha) = p$, zostrojíme prechody $\delta(q_0', \alpha) = p$.

- Vytvoríme nový akceptačný stav q_F. Prechody do neho zostrojíme podľa prechodov do pôvodných akceptačných stavov. Tj. pre všetky $\alpha \in \Sigma$ a pre všetky $q_x \in F$ (v pôvodnom NKA) ak existuje také $q_y \in Q$, že $\delta(q_y, \alpha) = q_x$, zostrojíme prechod $\delta(q_y, \alpha) = q_F$.

- Stavy nového NKA budú (pôvodné) $Q \cup \{q_0', q_F\}$. A množina akceptačných stavov bude obsahovať iba $\{q_F\}$.

Táto konštrukcia sa zvykne nazývať ako prasiatková. Resp. výsledný NKA je v takzvanom prasiatkovom normálnom tvare. Prečo? Pozrite sa na obrázok.



Riešenie príkladu 2b

Ako vieme túto konštrukciu použiť na riešenie podúlohy (b)? Skúsme si načrtnúť riešenie pre jazyk $L_1^2 = L_1L_1$. Keby sme zobrali prerobený automat pre jazyk v predchádzajúcej časti, stačilo by nám ho zduplikovať a dať za seba. Ako by sme ho spojili? Akceptačný stav prvej kópie a vstupný stav druhej kópie by sme zlúčili. Čo to spraví? Prvá časť slova, ktorá bude z jazyka L_1 sa prečíta v prvej kópii automatu a skončí v stave q_{F1}, následne sa začne čítať druhá časť slova v druhej kópii automatu a preto z q_{F1} prechádzame priamo doňho a skončíme v q_{F2}.

Skúste si premyslieť, ako takto skonštruovať ďalšie operácie na jazykoch. Ktoré zostrojiť idú a ktoré nie?

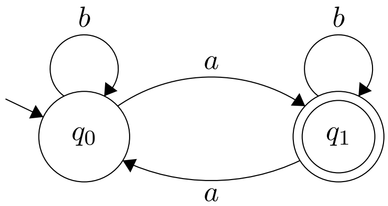
3. Príklad

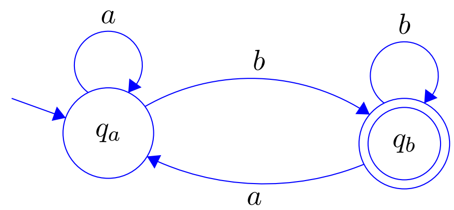
Nech $L_1 = \{w \in \{a,b\}^* \mid |w|_a \ mod\ 2 = 1\}$ a $L_2 = \{wb \mid w \in \{a,b\}^*\}$.

Zostrojte automat, ktorý bude rozoznávať jazyk $L_3 = \{xzy \mid x, y, z \in \{a,b\}^*, xy \in L_1 \land z \in L_2\}$

Riešenie 3. príkladu

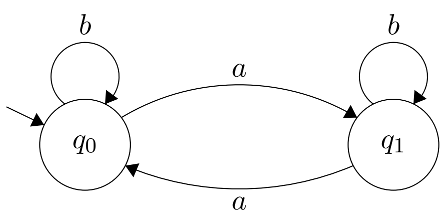
Zostrojme si automaty pre jazyky L_1 a L_2. Čierny automat A rozoznáva L_1 a modrý automat B rozoznáva L_2.

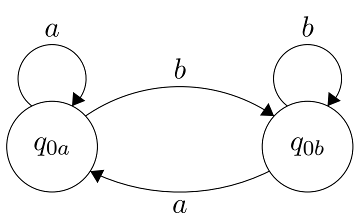


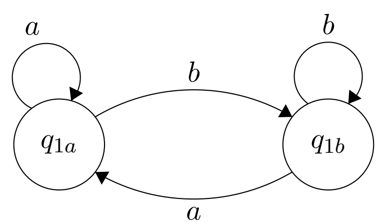


Ako teraz zostrojíme automat pre L_3? Pomôže nám tu prasiatkový tvar? Nie úplne, keďže my sa z čierneho automatu budeme musieť dostať do modrého aj z neakceptačného stavu. Ako ale v modrom automate budeme vedieť odkiaľ sme sa tam dostali? A ako budeme vedieť, kde máme pokračovať vo výpočte na y? Možno nám môže pomôcť niekoľko kópií každého automatu.

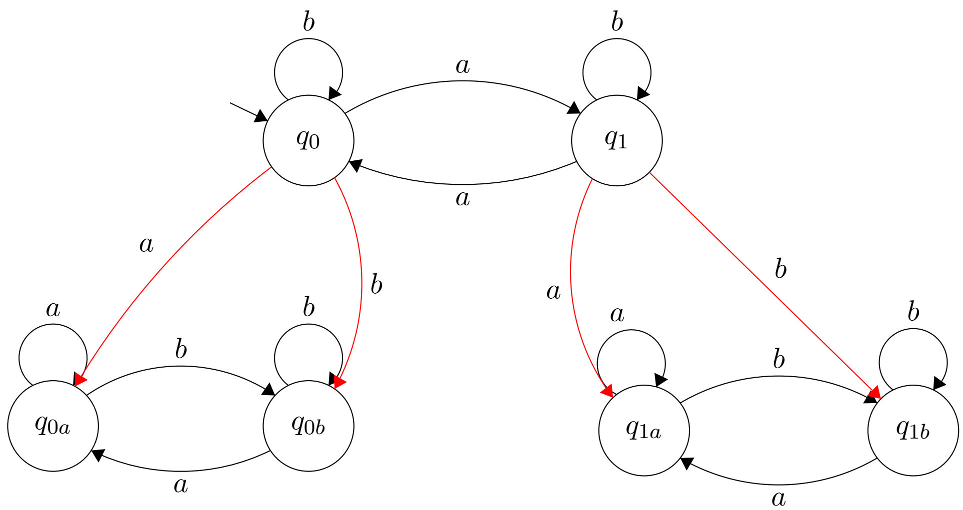
Poďme postupne. Automat A má dva stavy, čítanie x tak môže skončiť buď v stave q_0 alebo q_1. Potrebujeme si pamätať, kde sme skončili čítať x, aby sme dobre vedeli určiť, či xy je z jazyka L_1. Pre každý stav si preto vytvorme kópiu automatu B. Označme si stavy v týchto kópiách tak, aby sme vedeli, ako sme sa do nich dostali. Všimnite si, že sme zatiaľ nemáme žiaden stav označený ako akceptačný, keďže to budeme riešiť až v poslednom kroku.







Ako spraviť prechody z automatu A do jednotlivých kópií automatu B? Už sme naznačili, že prechod vždy pôjde z jedného stavu automatu A, a keďže týmto prechodom pôjde už nové slovo $z \in L_2$, tak sa pozeráme na prechody zo vstupného stavu automatu B, čiže q_a.



Vidíme, že zo stavu q_0 sme sa po a dostali do q_{0a}. Prečo? Keď v automate B príde na vstup a, dostaneme sa do q_a. Keď zas v automate B príde na vstup b, dostane sa do q_b, preto sme sa z q_0 dostali po b do q_{0b}. Keď sa na obrázok bližšie pozrieme, uvidíme, že prechody z q_1 sú rovnaké. Pretože tieto (červené) prechody sú pre podslová y z L_2.

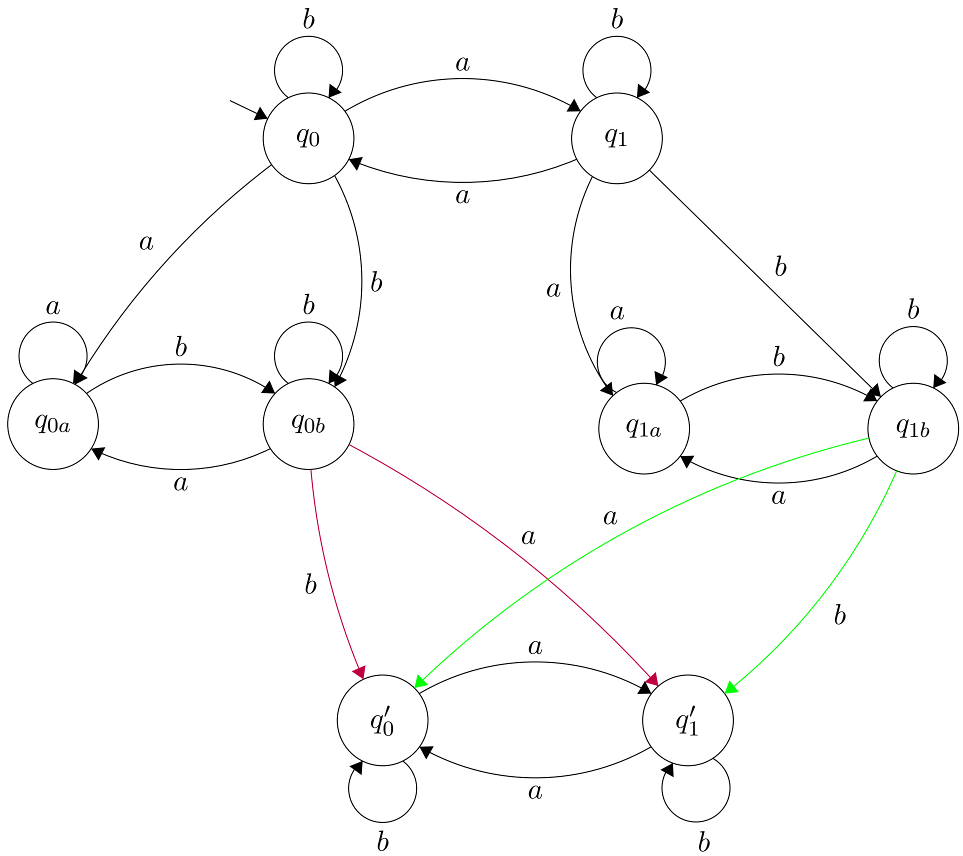
A čo ďalej? Kedy sa vieme posunúť z automatov B? Iba ak sme prečítali slovo $y \in L_2$, čiže ak skončíme v akceptačnom stave q_b. Dobre, budeme sa posúvať iba z q_{0b} a q_{1b}, ale kam a ako? Do prvého automatu? Nie, lebo keby sme sa dostali späť do pôvodného automatu, mali by sme zabezpečené, že sme prečítali podslovo y? Resp. mohlo by sa stať, že by sme sa do nejakej kópie automatu B nedostali? Mohlo. A tiež by sa mohlo stať to, že z pôvodného automatu by sme prešli do automatu B, potom sa vrátili, a potom opäť prešli do automatu B atď? Áno. Rozpoznali by sme takto správne slová, alebo aj slová, ktoré do jazyka nepatria? No asi aj také, ktoré nie sú korektné. Takže sa do pôvodného automatu vrátiť nemôžeme. Preto opäť vytvoríme kópiu prvého automatu. Bude nám stačiť jeden? Áno, keďže v ňom už budeme končiť, nebudeme prechádzať nikam ďalej, takže si nepotrebujeme nič pamätať. Aby bolo jasné, že ide o kópiu, označme stavy čiarkou.

No a čo s prechodmi? Pri q_{0b} vieme, že podslovo x sme prestali čítať v stave q_0, takže v čítaní podslova y by sme mali pokračovať od q_0'. Čo to znamená? Že sa pozeráme na prechody z q_0'. Ak do q_0' príde b, ostaneme v ňom. Tj. $\delta(q_{0b}, b) = q_0'$. A ak tam príde a, posunieme sa do q_1', čiže $\delta(q_{0b}, a) = q_1'$. V obrázku sú tieto prechody znázornené fialovou.

Pri q_{1b} vieme, že podslovo x sme prestali čítať v stave q_1, takže v čítaní podslova y by sme mali pokračovať od q_1'. Ak v q_1' čítame b, zostaneme v

q_1', preto aj $\delta(q_{1b}, b) = q_1'$. Ak v q_1' čítame a, presunieme sa do q_0', preto $\delta(q_{1b}, a) = q_0'$. V obrázku sú tieto prechody znázornené zelenou.

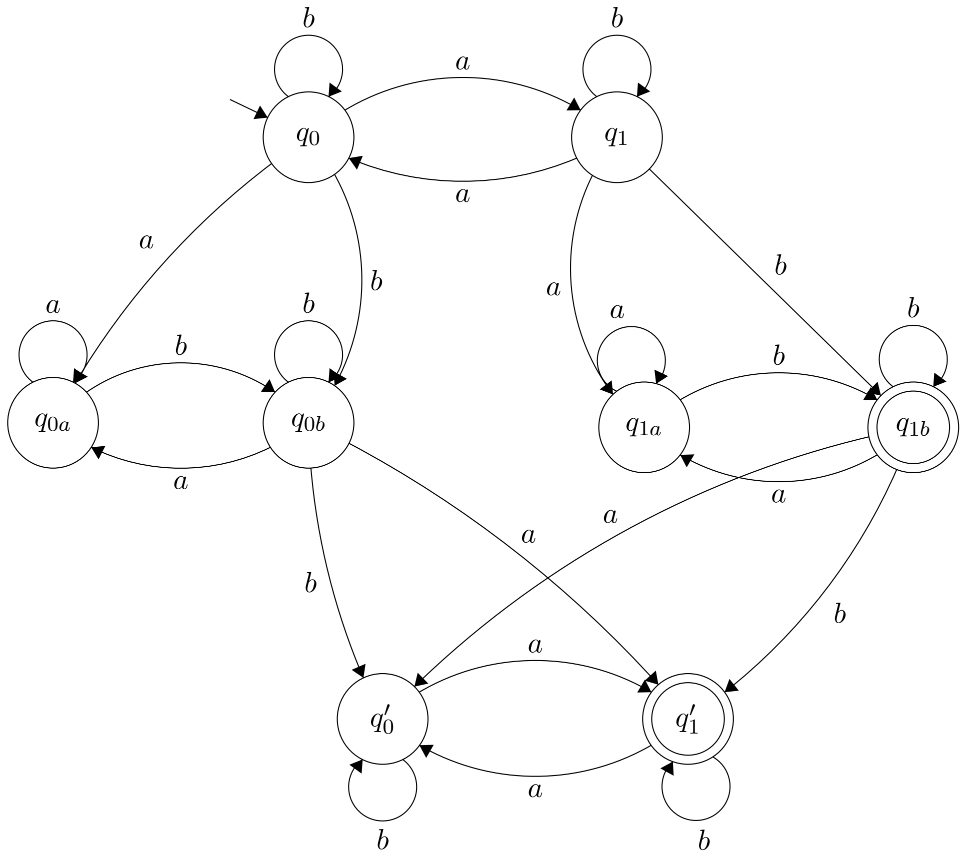
Zelené aj fialové prechody slúžia na čítanie podslova y.



Zhrňme si to. Prečítali sme podslovo x (prvý automat, na obrázku najvyššie). Prečítali sme podslovo z (v automatoch v strede), to bolo celé z jazyka L_2, preto sme sa posúvali iba zo stavov q_{0b} a q_{1b}. No a prečítali sme aj podslovo y, pričom sme si dali pozor na to, aby $xy \in L_1$. Ktorý stav bude teda akceptačný?

Určite to bude q_1'. Ale môže byť aj nejaký iný stav akceptačný? Keď sa pozrieme na zadanie, zistíme, že $x, y, z \in \{a, b\}^*$, čiže môžu byť aj λ. Všetky? $\lambda \notin L_2$, takže $z \neq \lambda$. To, že $x = \lambda$ máme zaručené prechodmi z q_0, kde sa hneď môže čítať podslovo z. A čo ak $y = \lambda$? To by znamenalo, že skončíme už v pôvodnom akceptačnom stave niektorej kópie druhého automatu. Ako zistíme ktorej? Musí platiť, že $x\lambda \in L_1 \rightarrow x \in L_1$. Teda v prvom automate sme museli skončiť v akceptačnom stave, tj. q_1. Ďalší akceptačný stav tak bude stav q_{1b}

Výsledný automat:



4. príklad

Majme jazyky:

- $L_1 = \{w \in \{0,1\}^* \mid |w|_0 \; mod \; 3 = 1\}$
- $L_2 = \{w \in \{0,1\}^* \mid |w|_0 \; mod \; 3 = 2\}$
- $L_3 = \{w \in \{0,1\}^* \mid |w|_0 \; mod \; 3 = 0\}$
- $L_k = \{w0 \mid w \in \{0,1\}^*\}$
- $L_z = \{1w \mid w \in \{0,1\}^*\}$

Vyskúšajte zostrojiť automaty pre jazyky, ak $x, y, z, w \in \{0,1\}^*$

(a) $L_a = xzy \mid xy \in L_1, z \in L_k$

(b) $L_b = xzy \mid xy \in L_k, z \in L_2$

(c) $L_c = xzy \mid xy \in L_3, z \in L_z$

(d) $L_d = xzy \mid xy \in L_k, z \in L_z$

(e) $L_e = xzy \mid xy \in L_2, z \in L_3$

(f) $L_f = xzyw \mid xy \in L_k, zw \in L_z$

(g) $L_g = xzyw \mid xy \in L_1, zw \in L_k$

(h) $L_h = xzyw \mid xy \in L_2, zw \in L_3$

Koľko kópií automatov každého typu budete musieť spraviť? Čo všetko si budete musieť pamätať? Budú sa automaty pre L_a až L_e líšiť od automatov L_f až L_h počtom kópií automatov, a počtom stavov? Ktoré stavy budú akceptačné?

Zmenia sa výsledné automaty, ak $x, y, z, w \in \{0,1\}^+$?