

Sinchronizuojantys baigtiniai automatai

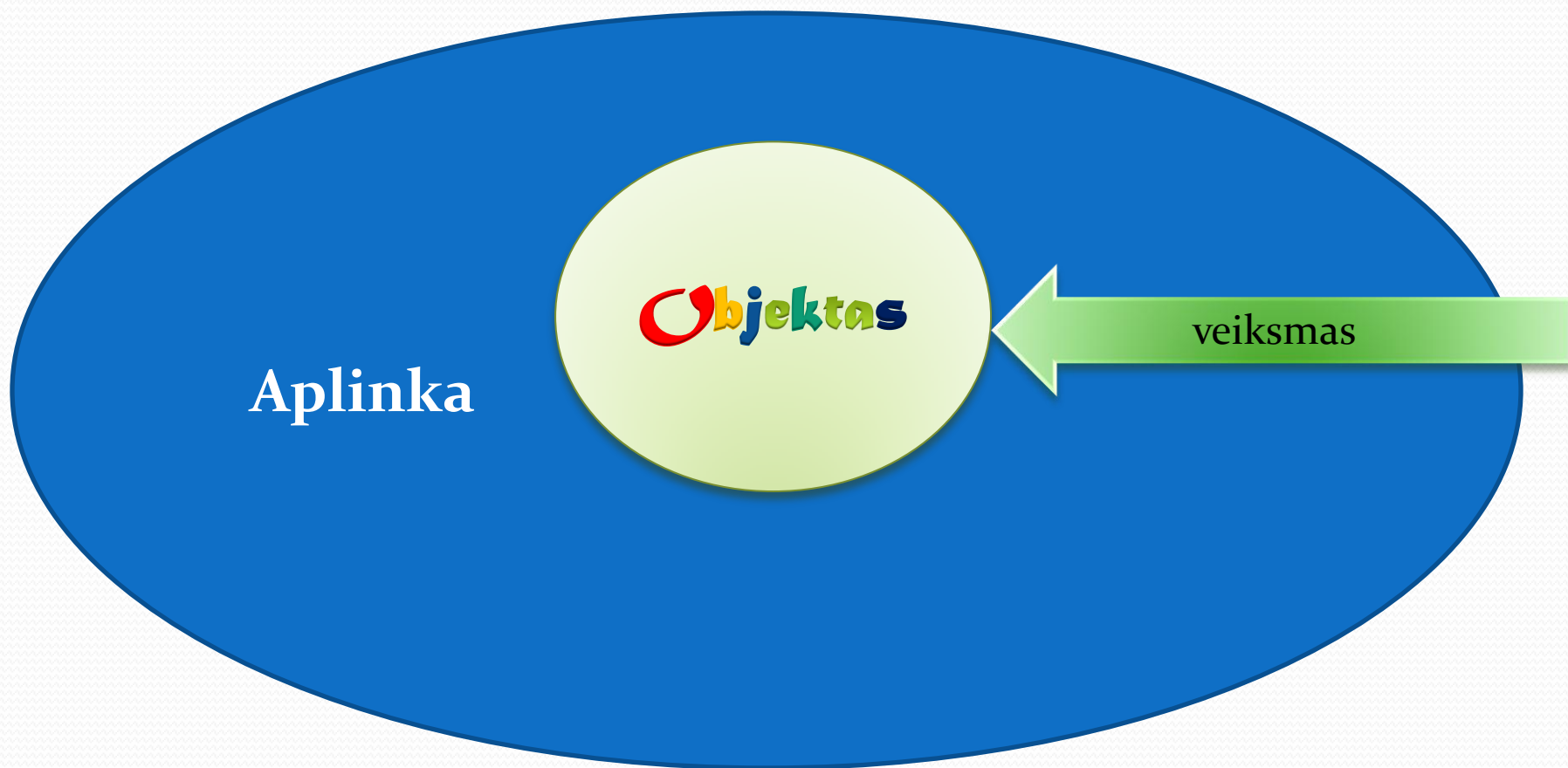
Rimantas Grigutis

Palanga 2012

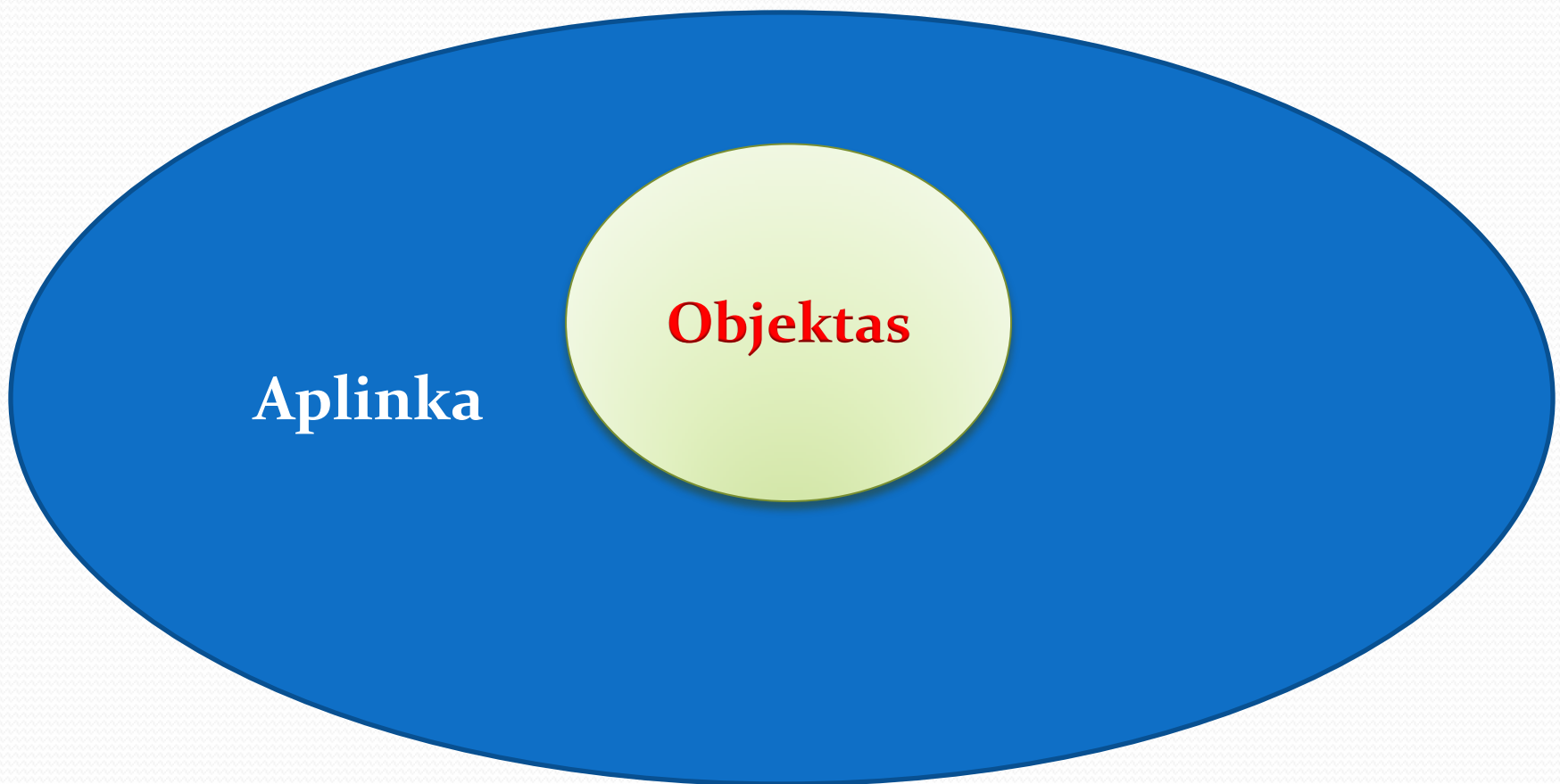
Baigtiniais automatais yra labai patogų realizuoti objekto ir aplinkos sąveiką



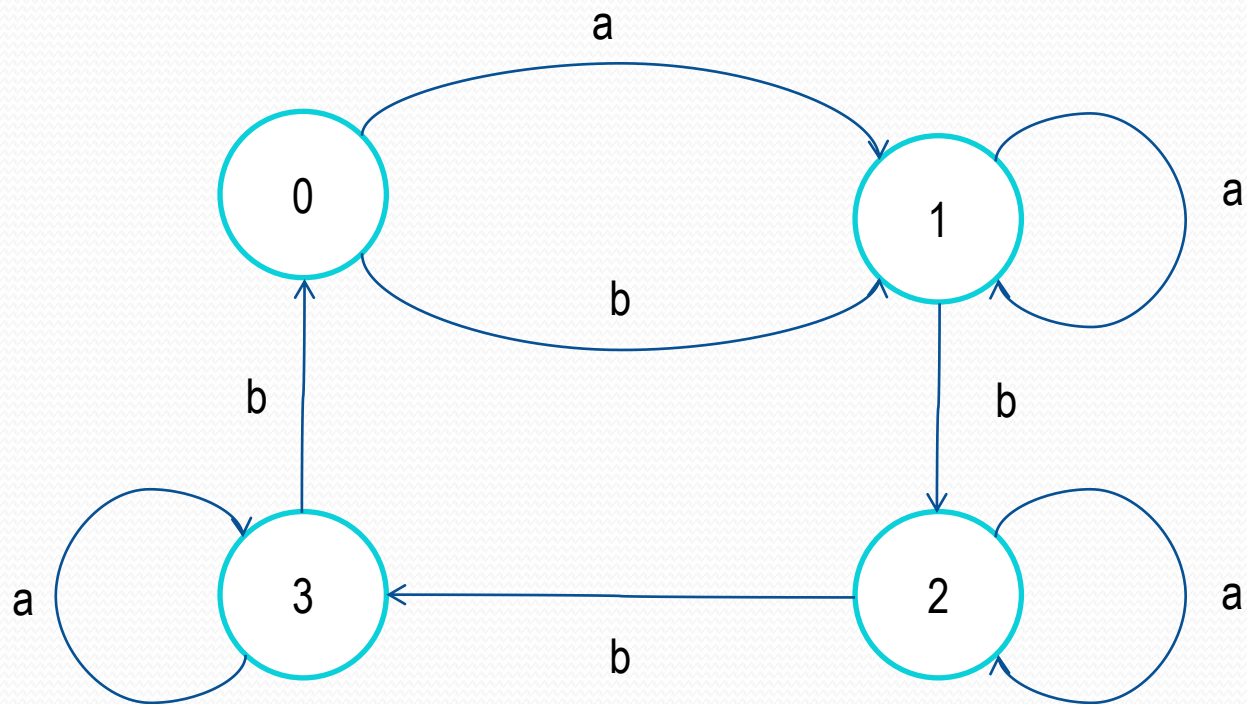
Baigtiniais automatais yra labai patogiu realizuojanti objekto ir aplinkos sąveiką



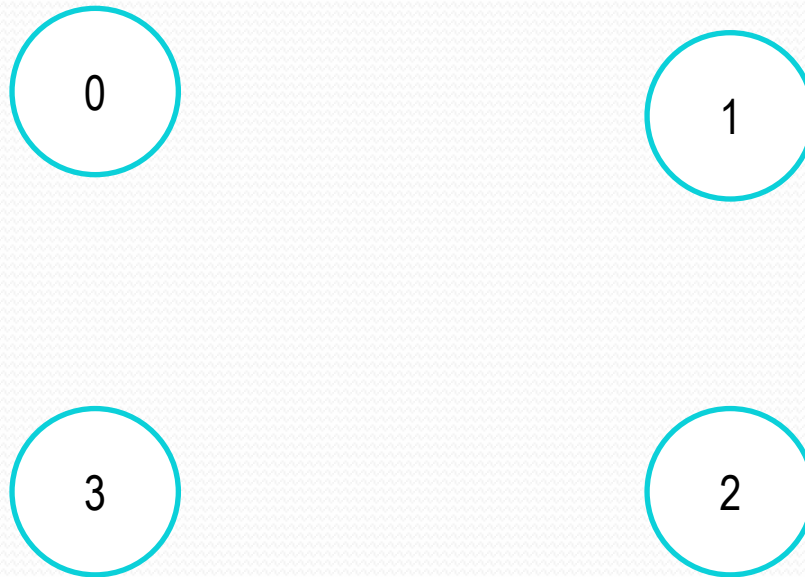
Baigtiniais automatais yra labai patogų realizuoti objekto ir aplinkos sąveiką



Baigtinį automatą patogiu vaizduoti diagrama

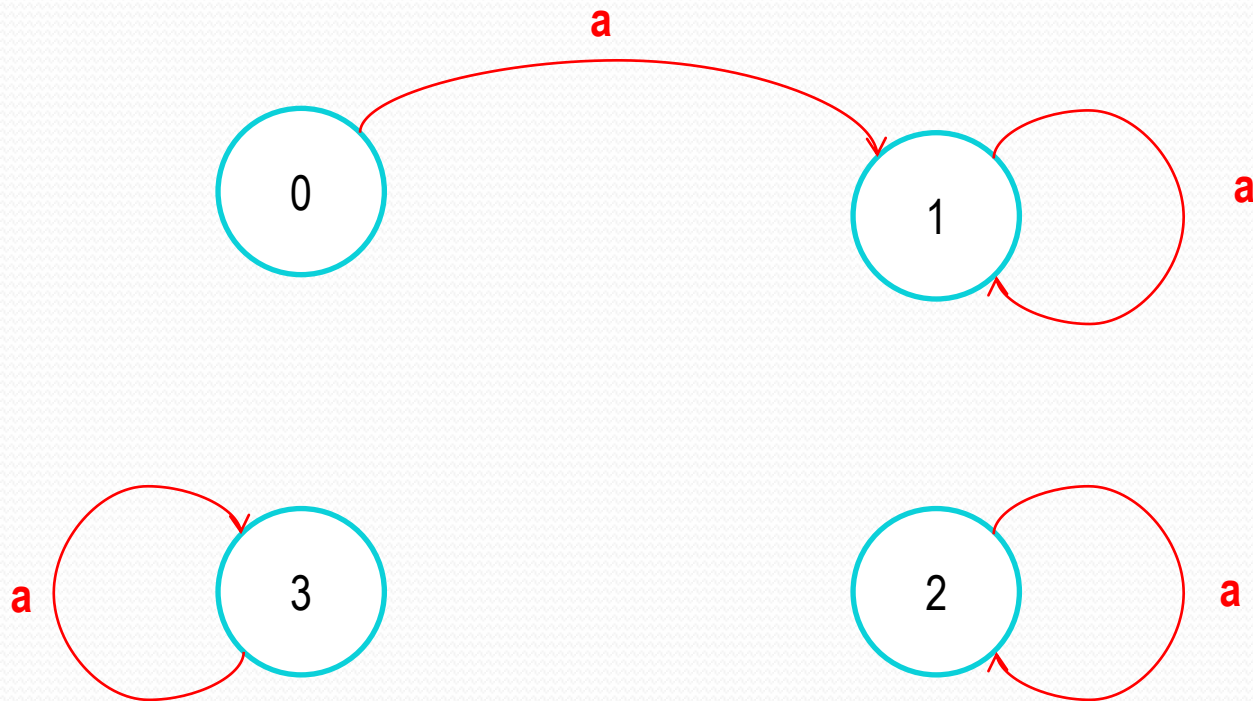


Baigtinį automatą patogiai vaizduoti diagrama



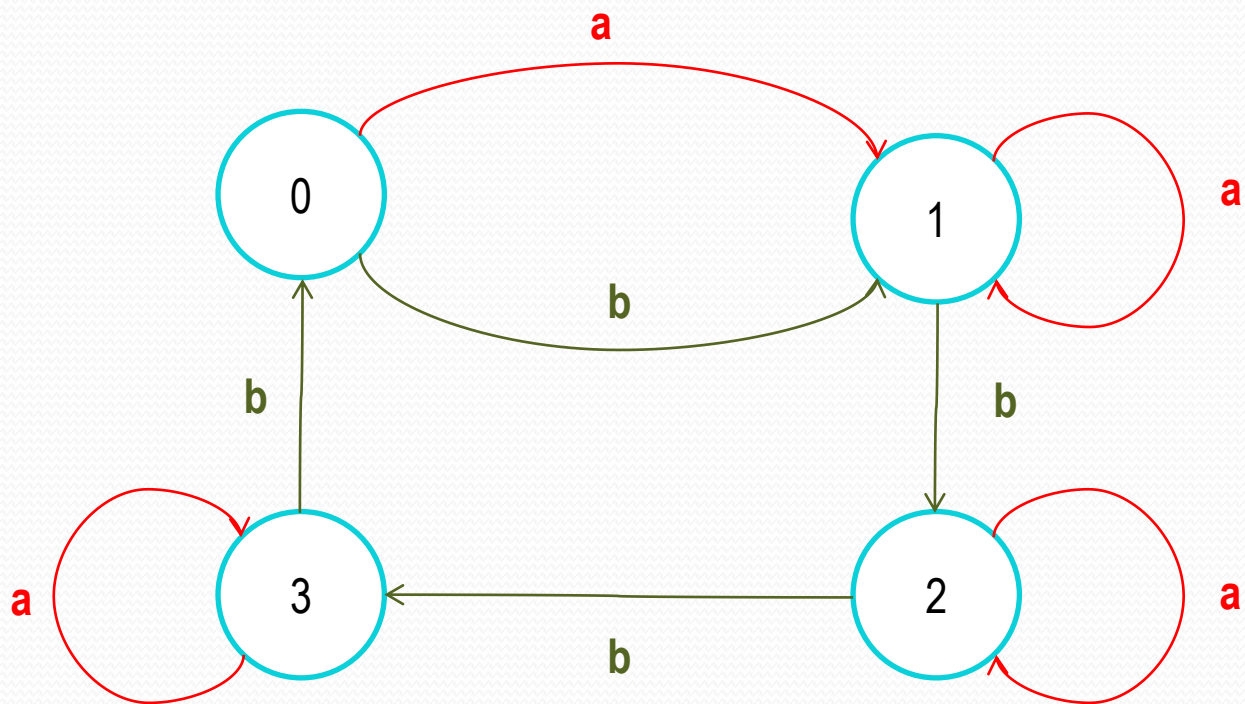
Čia matome 4 automato būsenas, pažymėtas 0,1,2,3

Baigtinį automatą patogiu vaizduoti diagrama



Čia matome 4 automato būsenas, pažymėtas 0,1,2,3
ir veiksmą pažymėtą raide **a**

Baigtinį automatą patogiu vaizduoti diagrama



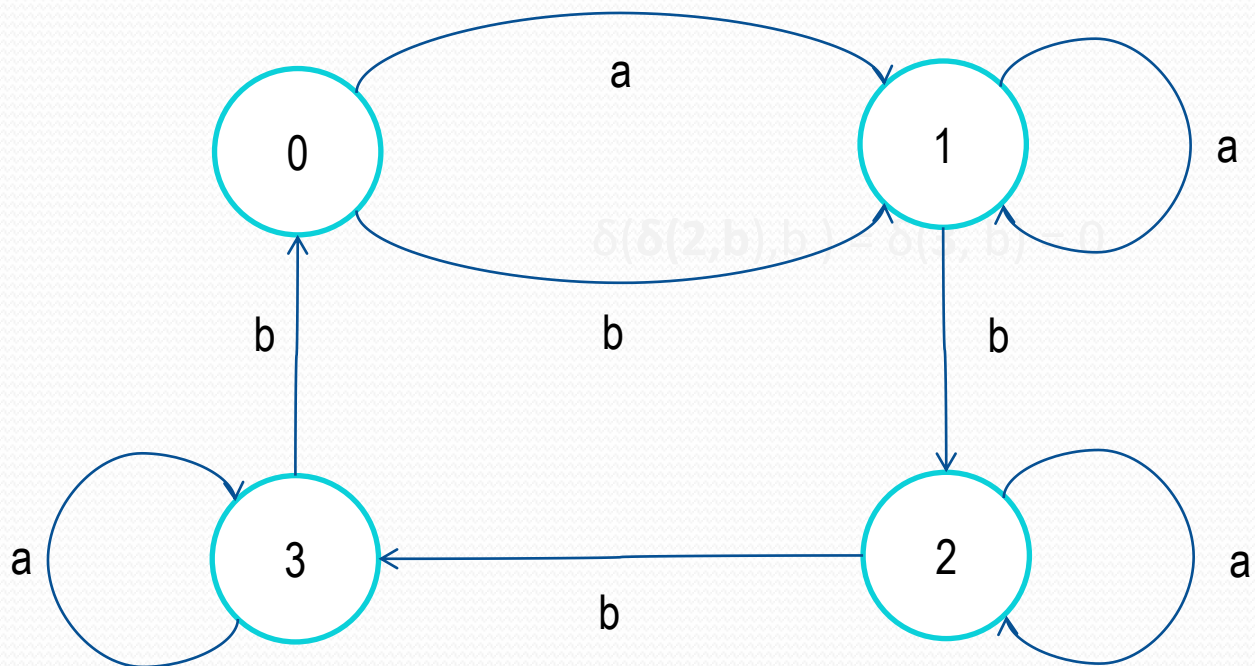
Čia matome 4 automato būsenas, pažymėtas 0,1,2,3,
vieną veiksmą pažymėtą raide **a** ir kitą veiksmą pažymėtą raide **b**

Mes nagrinėjame pilną determinuotą baigtinį automatą:

$$A = (Q, \Sigma, \delta)$$

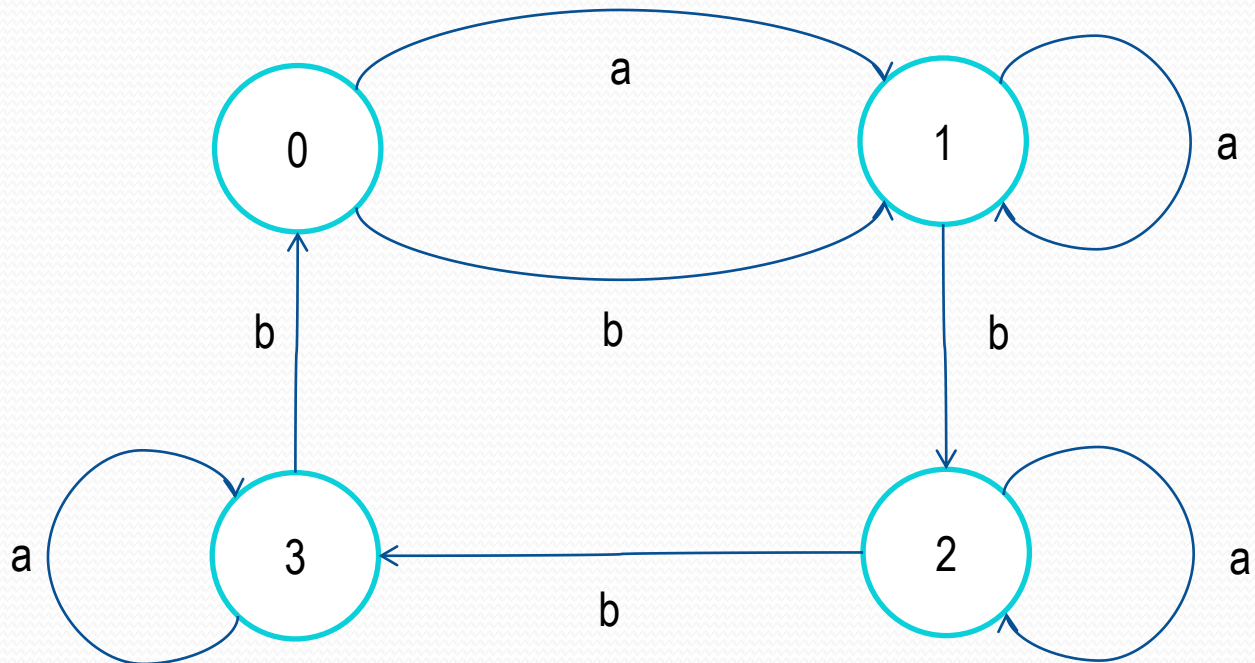
Čia

- Q yra automato baigtinė **būsenų aibė**;
- Σ yra automato baigtinė **abėcėlė**;
- $\delta: Q \times \Sigma \rightarrow Q$ yra automato visiškai apibrėžta **pėrėjimo** (*transition*) **funkcija**.



Funkcija δ natūraliai apibendrinama funkcija $\delta: Q \times \Sigma^* \rightarrow Q$.
Čia Σ^* žymi visų žodžių virš abėcėlės Σ , įskaitant ir tuščią žodį, aibę.
Pavyzdžiui,

$$\begin{aligned}\delta(1, abb) &= \\ \delta(\delta(1, a), bb) &= \delta(1, bb) = \\ \delta(\delta(1, b), b) &= \delta(2, b) = 3\end{aligned}$$



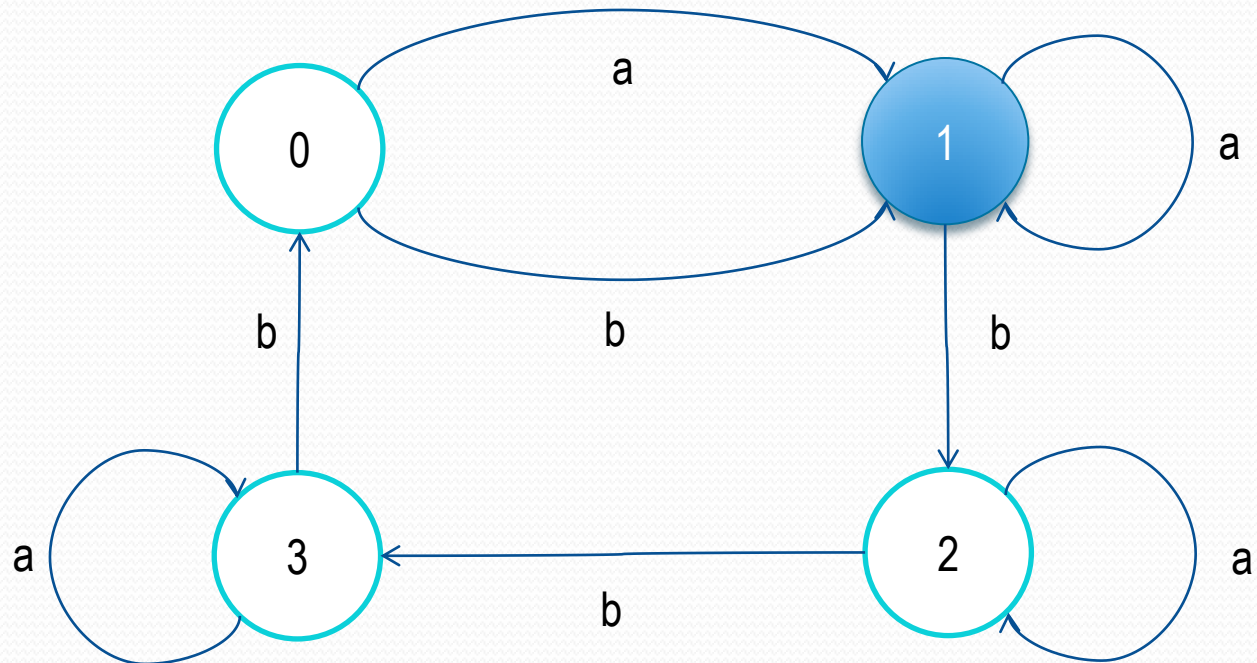
Funkcija δ natūraliai apibendrinama funkcija $\delta: Q \times \Sigma^* \rightarrow Q$.

Pavyzdžiui,

$$\delta(1, abb) =$$

$$\delta(\delta(1, a), bb) = \delta(2, bb) =$$

$$\delta(\delta(2, b), b) = \delta(3, b) = 0$$



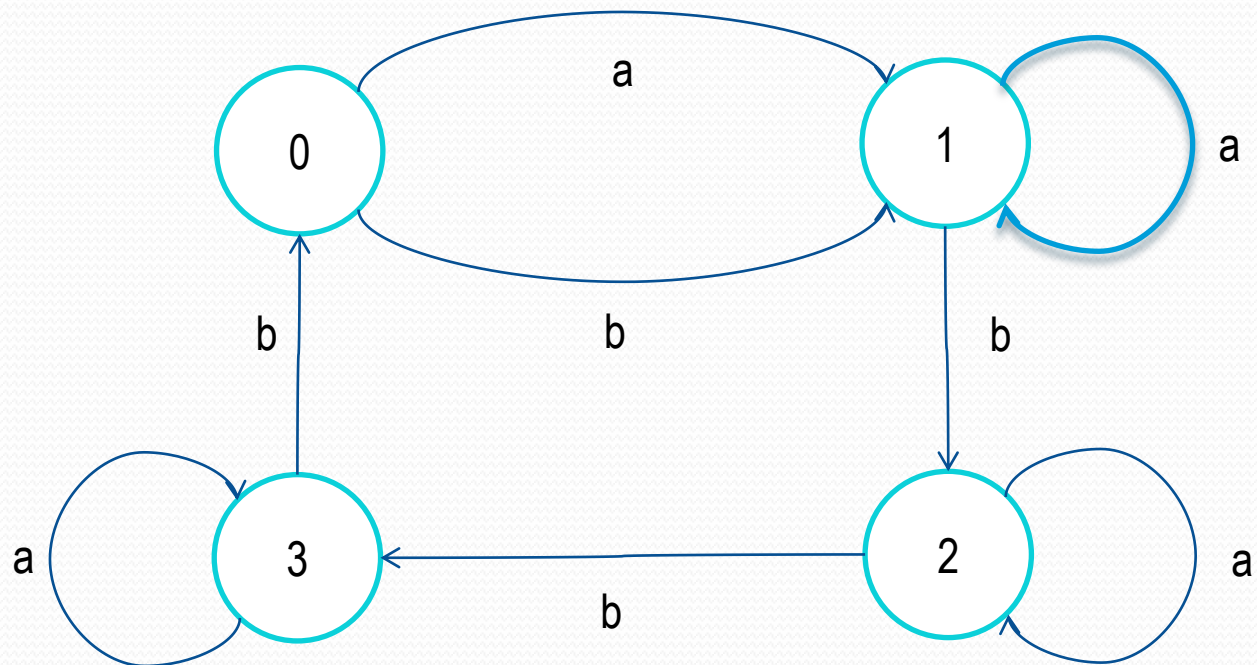
Funkcija δ natūraliai apibendrinama funkcija $\delta: Q \times \Sigma^* \rightarrow Q$.

Pavyzdžiui,

$$\delta(1, abb) =$$

$$\delta(\delta(1, a), bb) = \delta(1, bb) =$$

$$\delta(\delta(2, b), b) = \delta(3, b) = 0$$



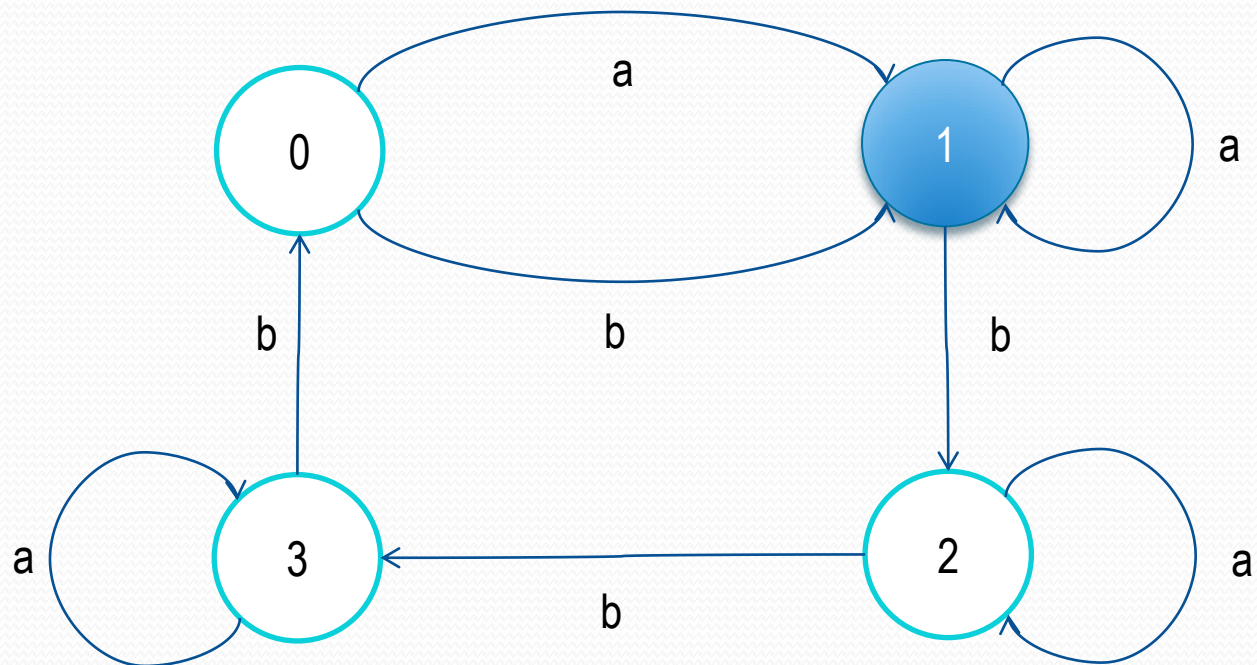
Funkcija δ natūraliai apibendrinama funkcija $\delta: Q \times \Sigma^* \rightarrow Q$.

Pavyzdžiui,

$$\delta(1, abb) =$$

$$\delta(\delta(1, a), bb) = \delta(1, bb) =$$

$$\delta(\delta(2, b), b) = \delta(3, b) = 0$$



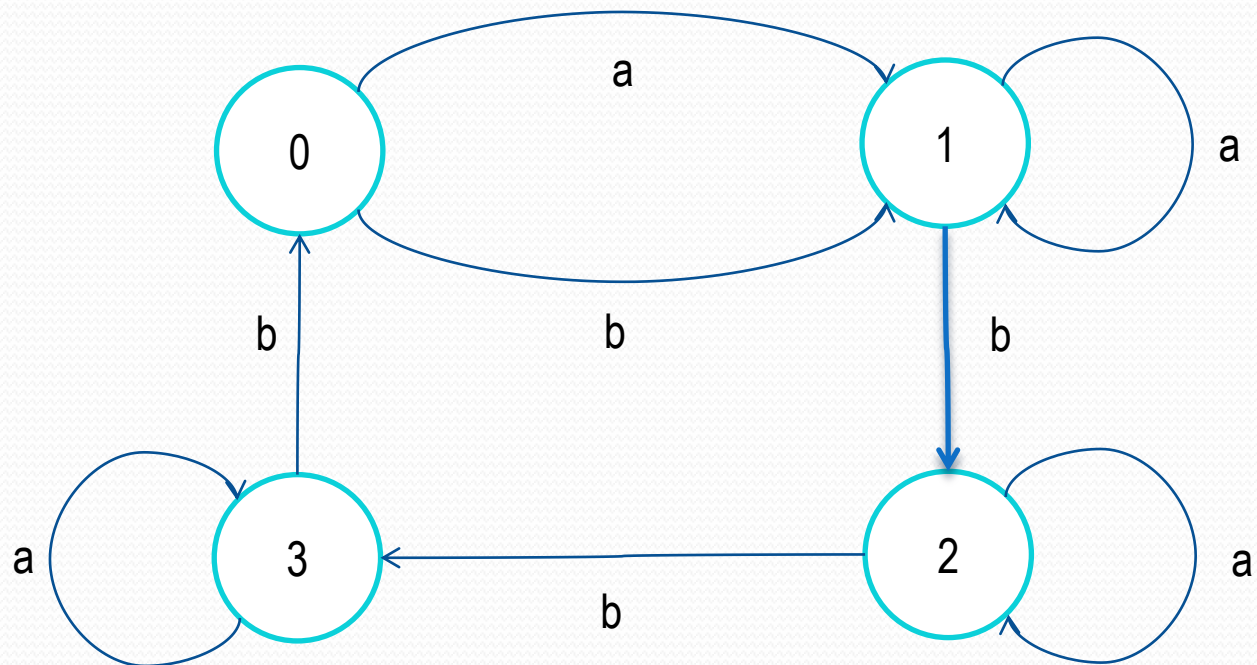
Funkcija δ natūraliai apibendrinama funkcija $\delta: Q \times \Sigma^* \rightarrow Q$.

Pavyzdžiui,

$$\delta(1, abb) =$$

$$\delta(\delta(1, a), bb) = \delta(1, bb) =$$

$$\delta(\delta(1, b), b) = \delta(2, b) = 3$$



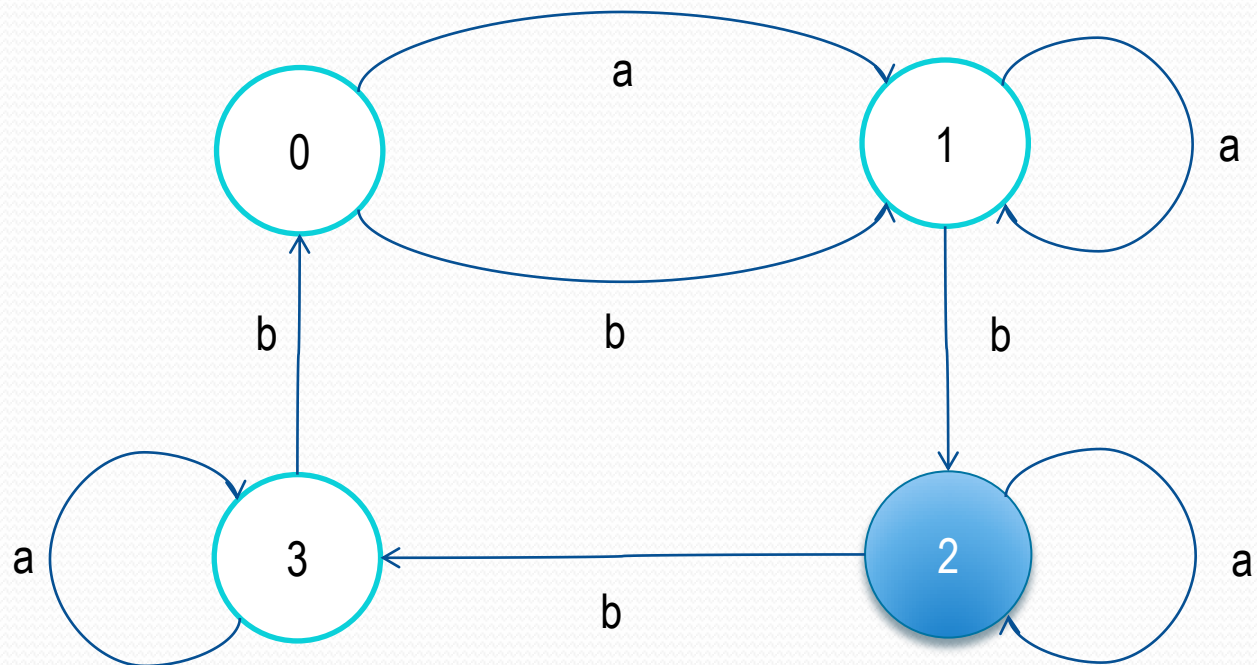
Funkcija δ natūraliai apibendrinama funkcija $\delta: Q \times \Sigma^* \rightarrow Q$.

Pavyzdžiui,

$$\delta(1, abb) =$$

$$\delta(\delta(1, a), bb) = \delta(1, bb) =$$

$$\delta(\delta(1, b), b) = \delta(2, b) = 3$$



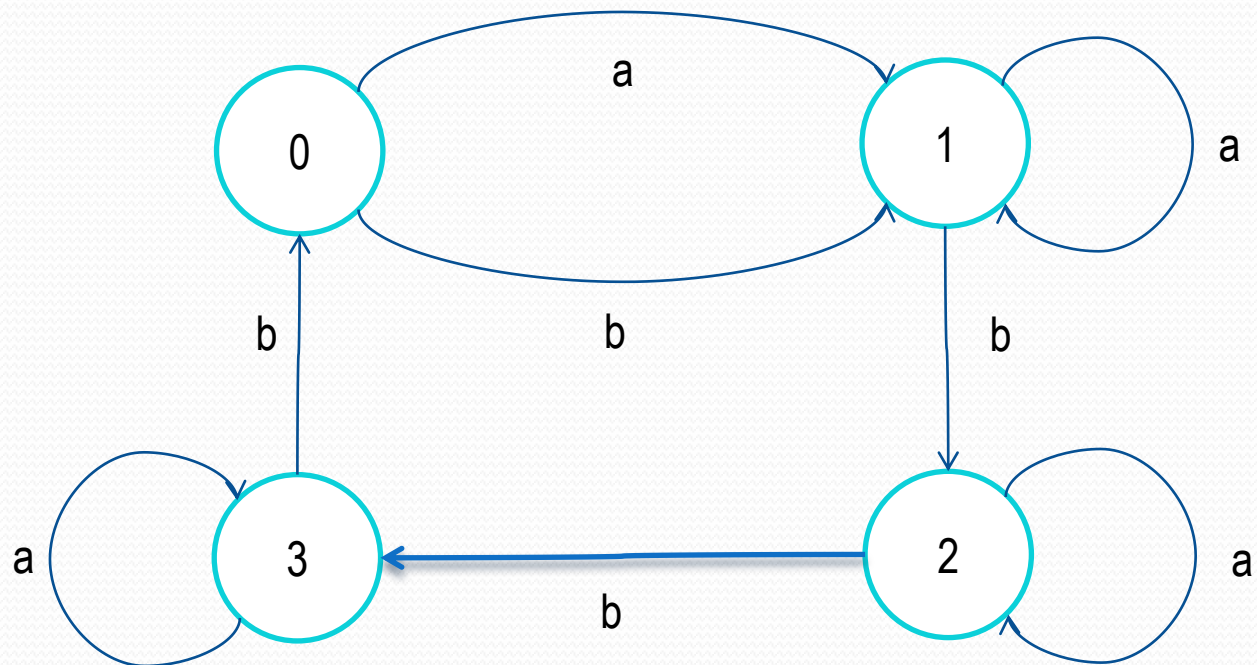
Funkcija δ natūraliai apibendrinama funkcija $\delta: Q \times \Sigma^* \rightarrow Q$.

Pavyzdžiui,

$$\delta(1, abb) =$$

$$\delta(\delta(1, a), bb) = \delta(1, bb) =$$

$$\delta(\delta(1, b), b) = \delta(2, b)$$



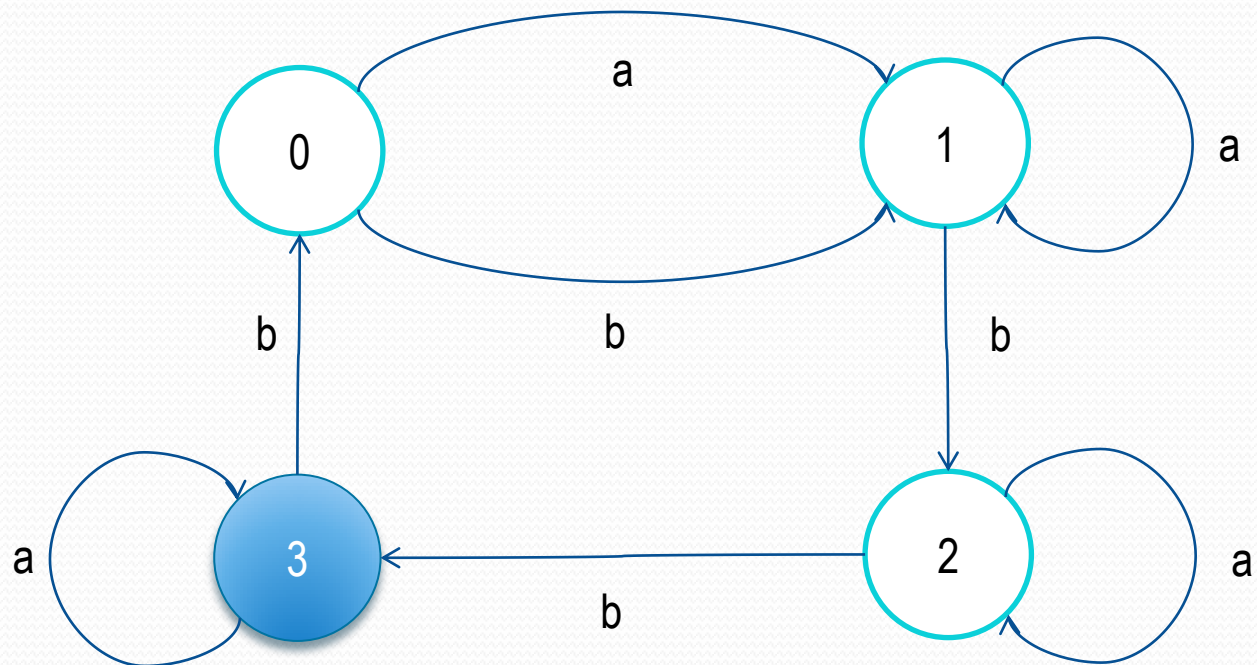
Funkcija δ natūraliai apibendrinama funkcija $\delta: Q \times \Sigma^* \rightarrow Q$.

Pavyzdžiui,

$$\delta(1, abb) =$$

$$\delta(\delta(1, a), bb) = \delta(1, bb) =$$

$$\delta(\delta(1, b), b) = \delta(2, b) = 3$$

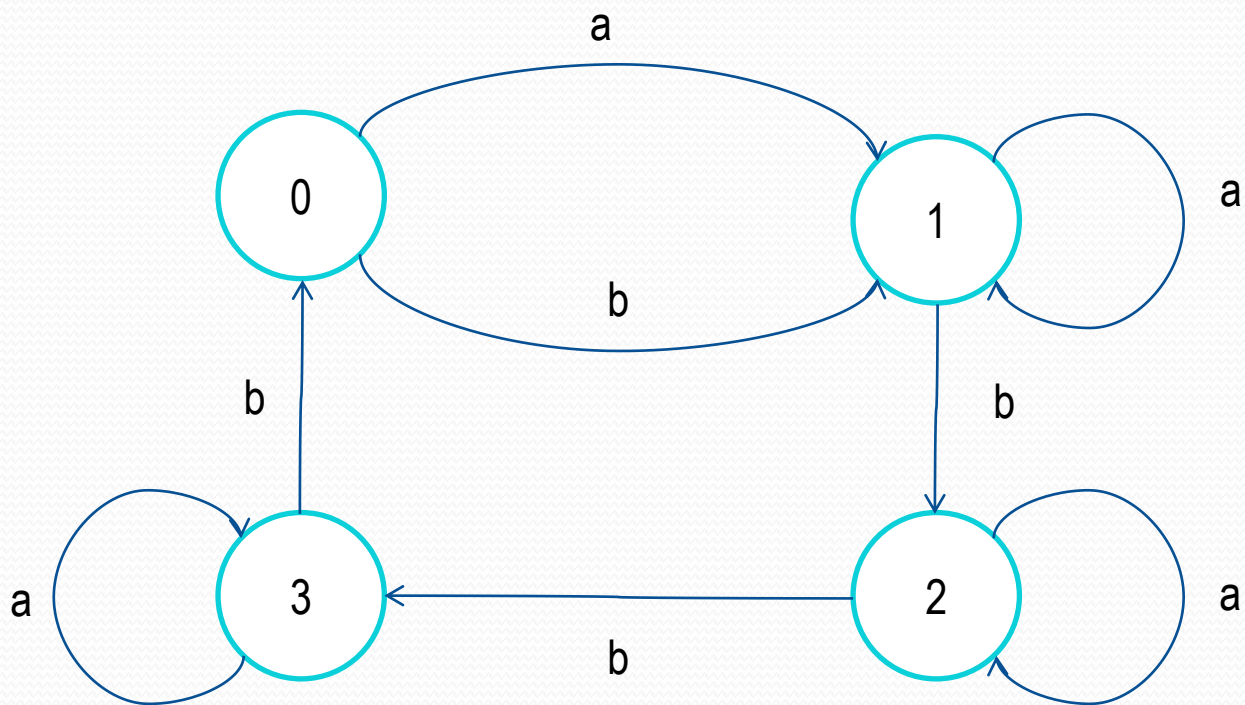


Automatas $A = (Q, \Sigma, \delta)$ vadinamas **sinchronizuojančiu**,
jei egzistuoja toks žodis $w \in \Sigma^*$,
kuris automata A perveda į **vienintelę**, iš anksto žinomą būseną q_0 ,
nepriklausomai nuo to, kurioje būsenoje prieš tai buvo automatas A :

$$\delta(q, w) = \delta(q', w) = q_0 \quad \text{su visais } q, q' \in Q.$$

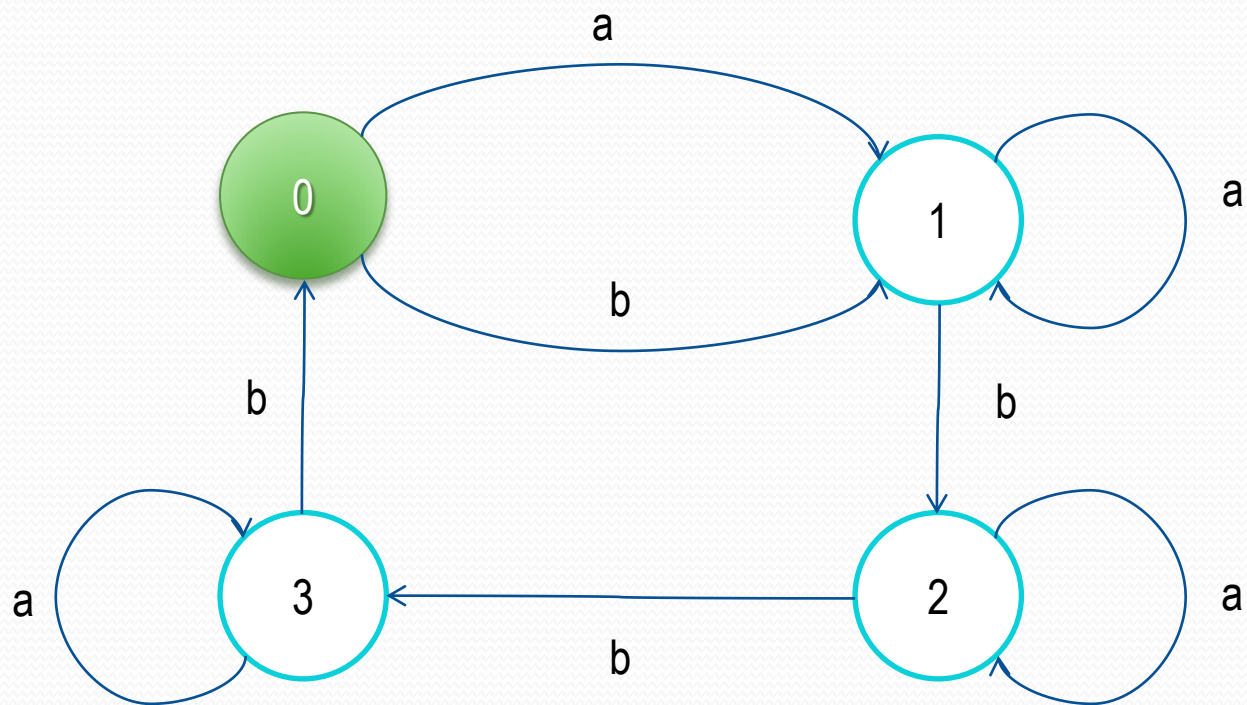
Žodis w tenkinantis šią savybę vadinamas *automatą A sinchronizuojančiu žodžiu* (ang. *reset word*).

Sinchronizuojančio automato pavyzdys



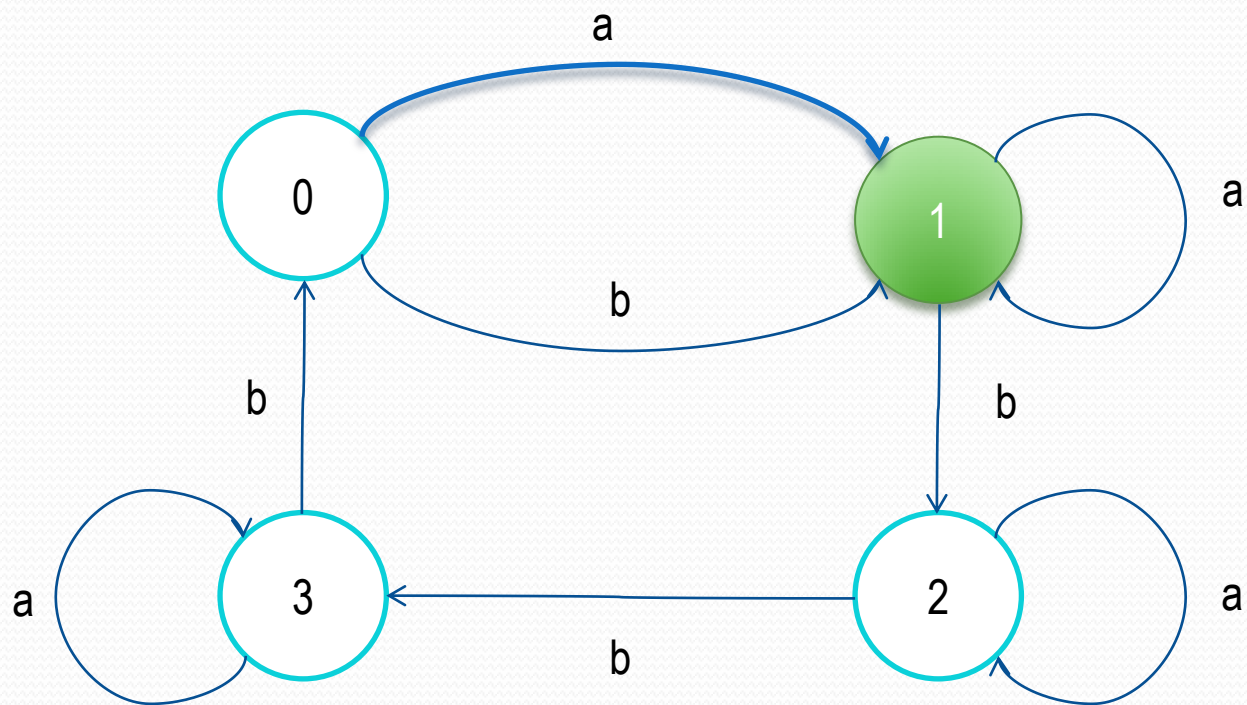
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



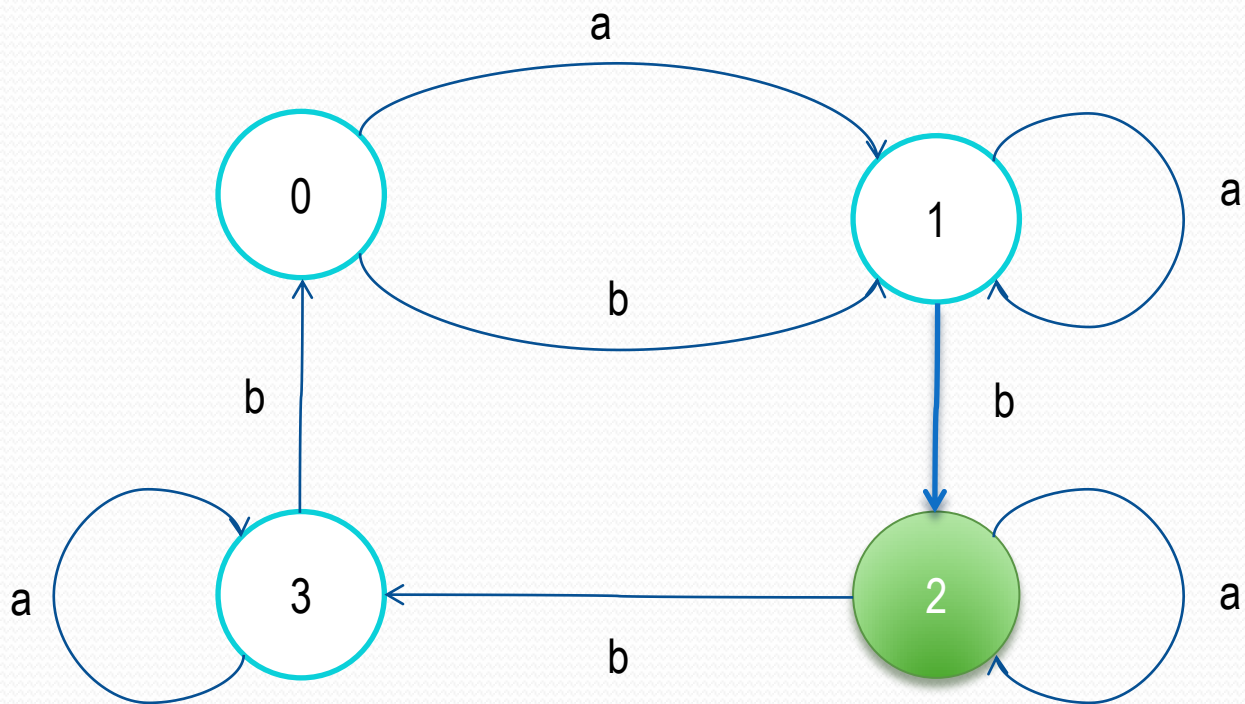
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



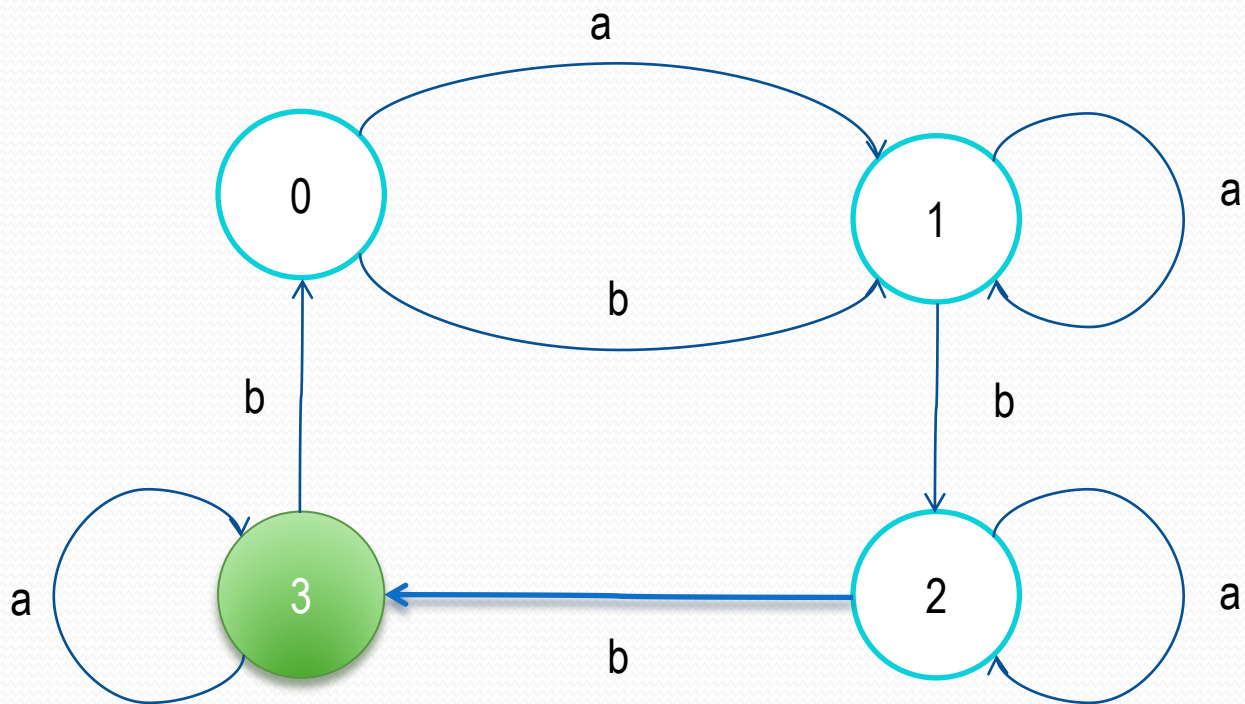
Šį automatą sinchronizuojantis žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



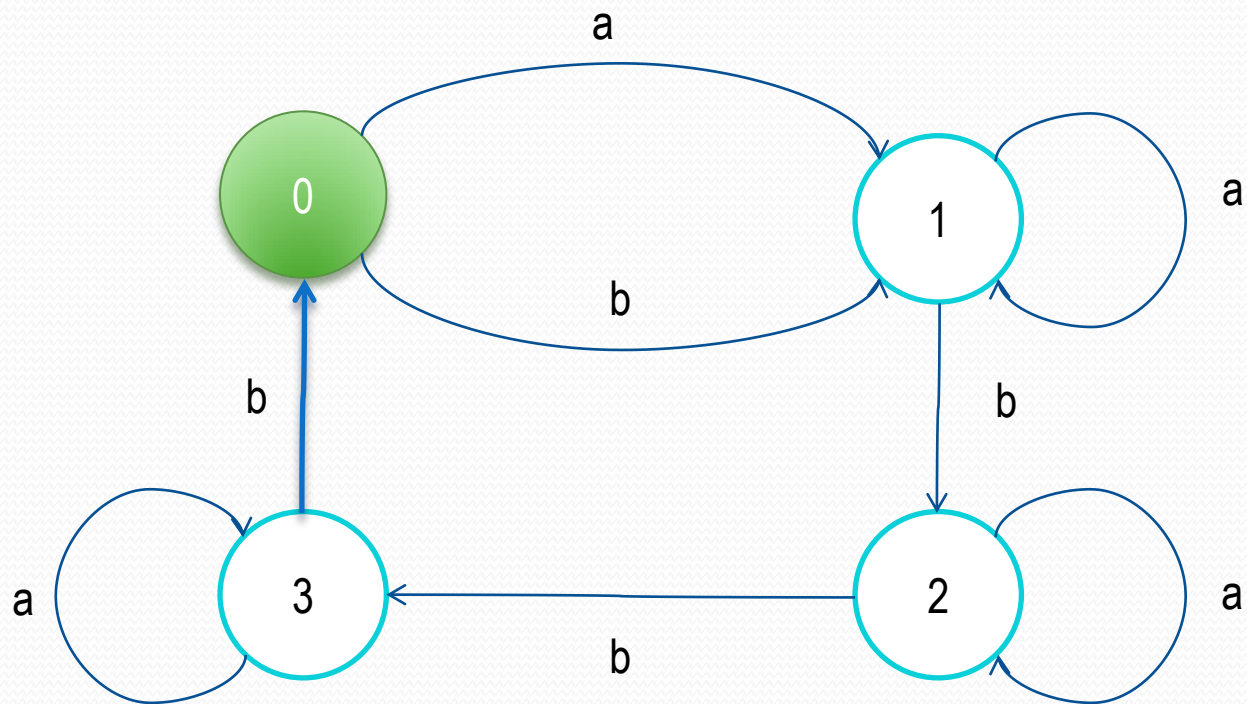
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



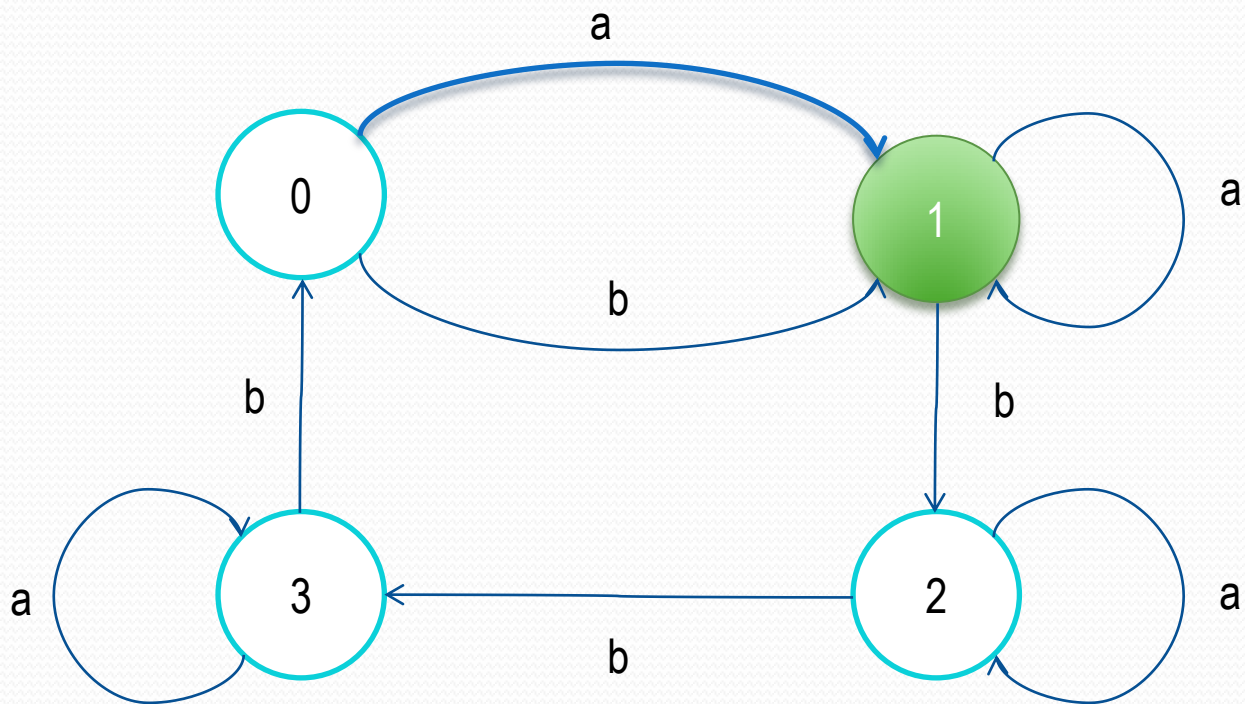
Šį automatą sinchronizuojantis žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



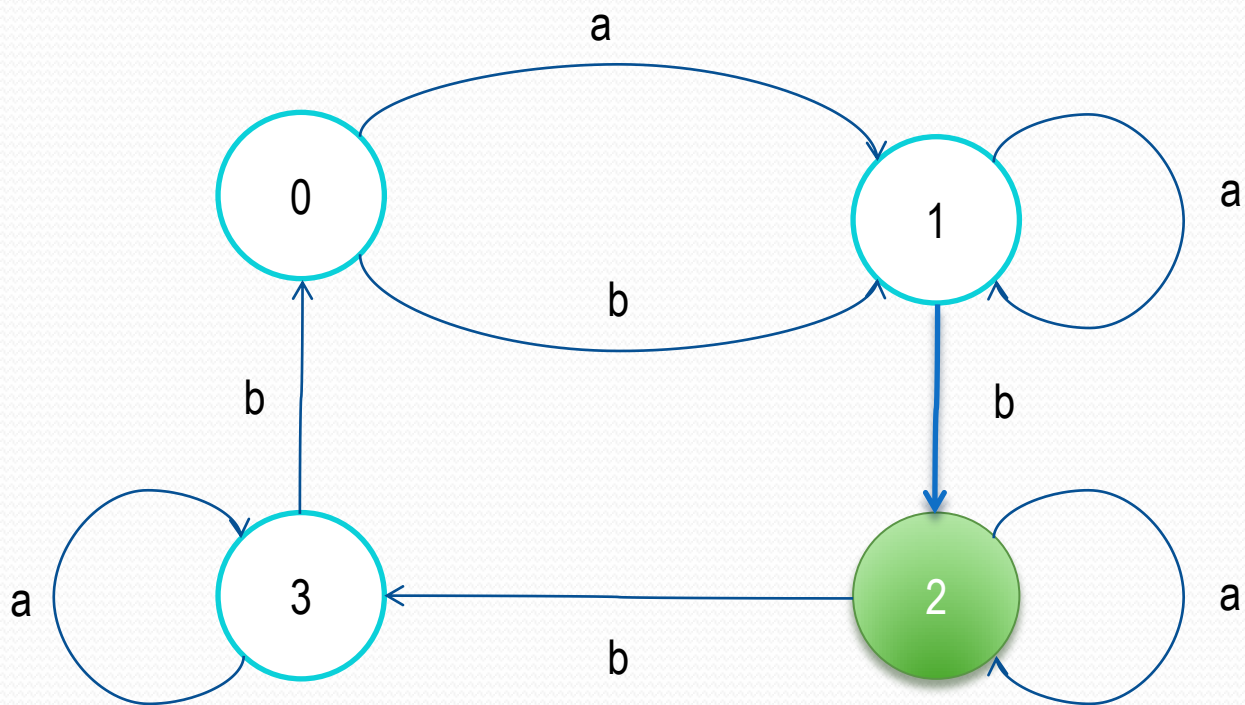
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



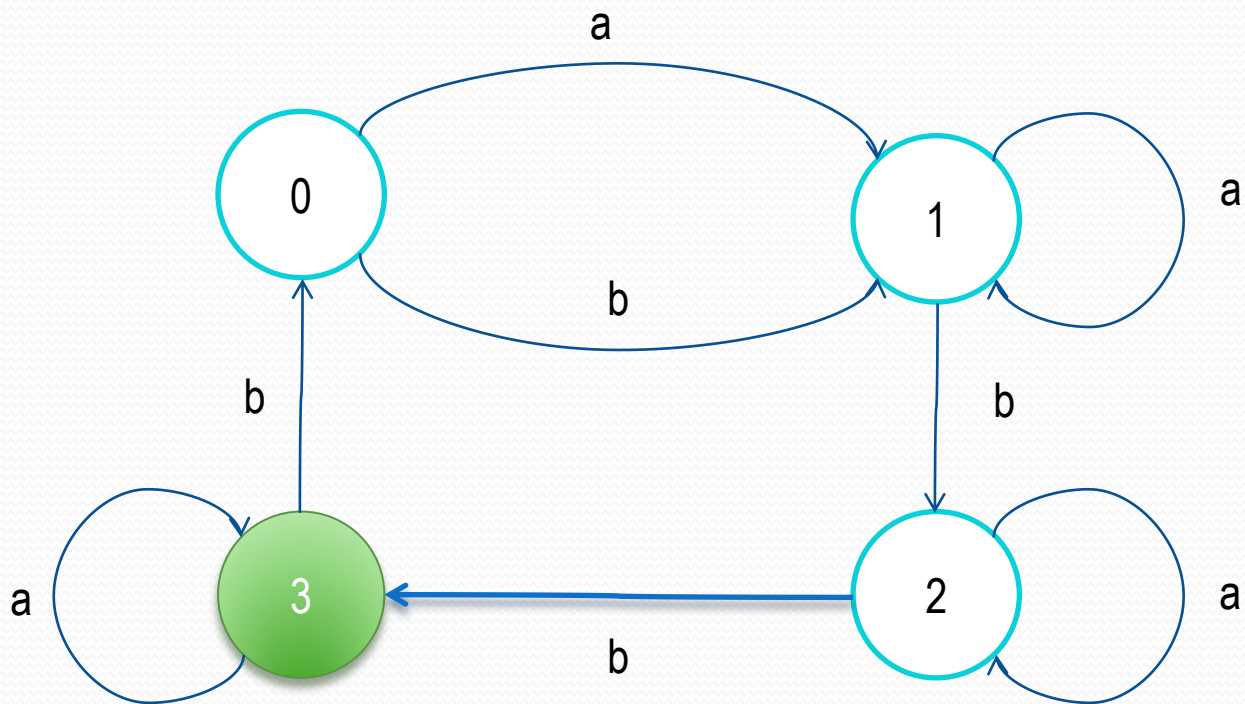
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



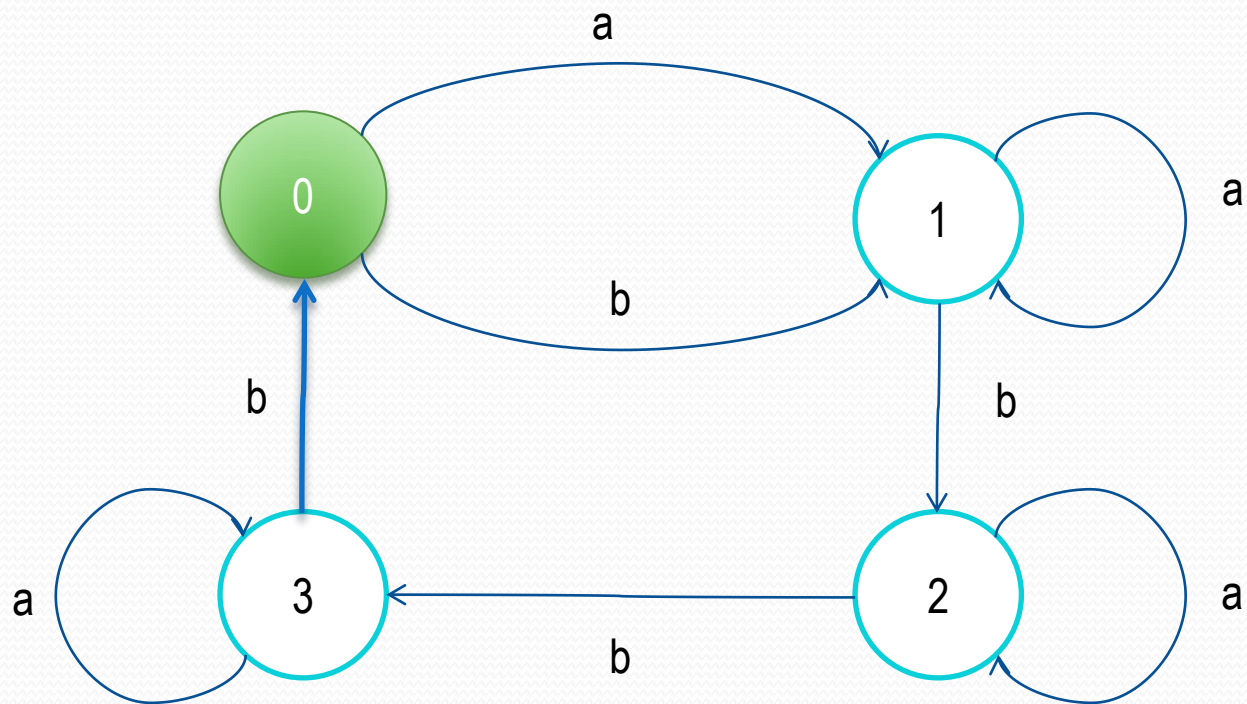
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



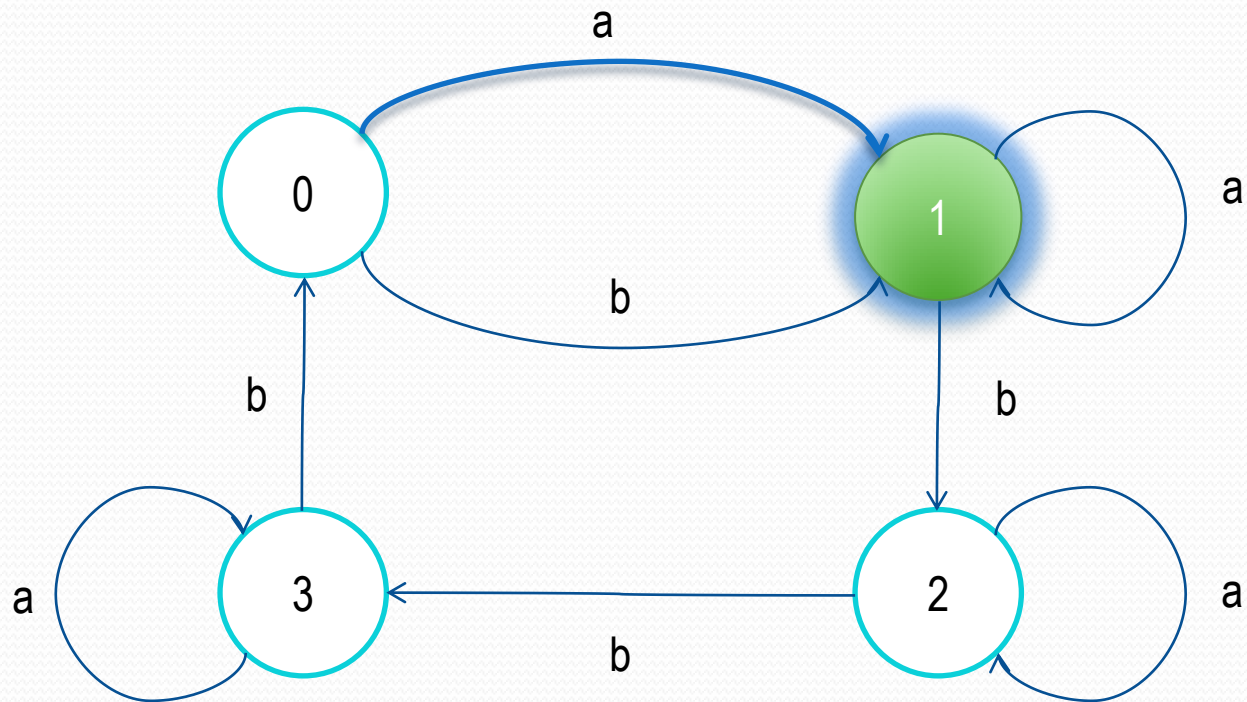
Šį automatą sinchronizuojantis žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



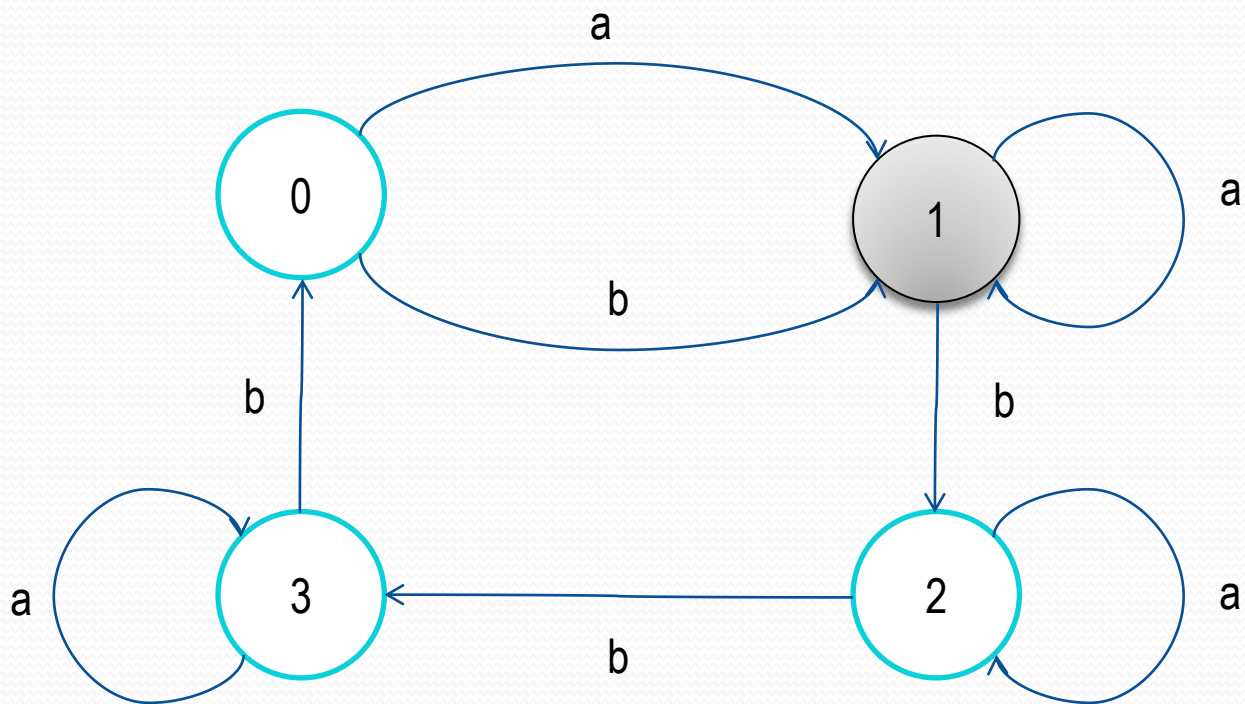
Šį automatą sinchronizuojantis žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



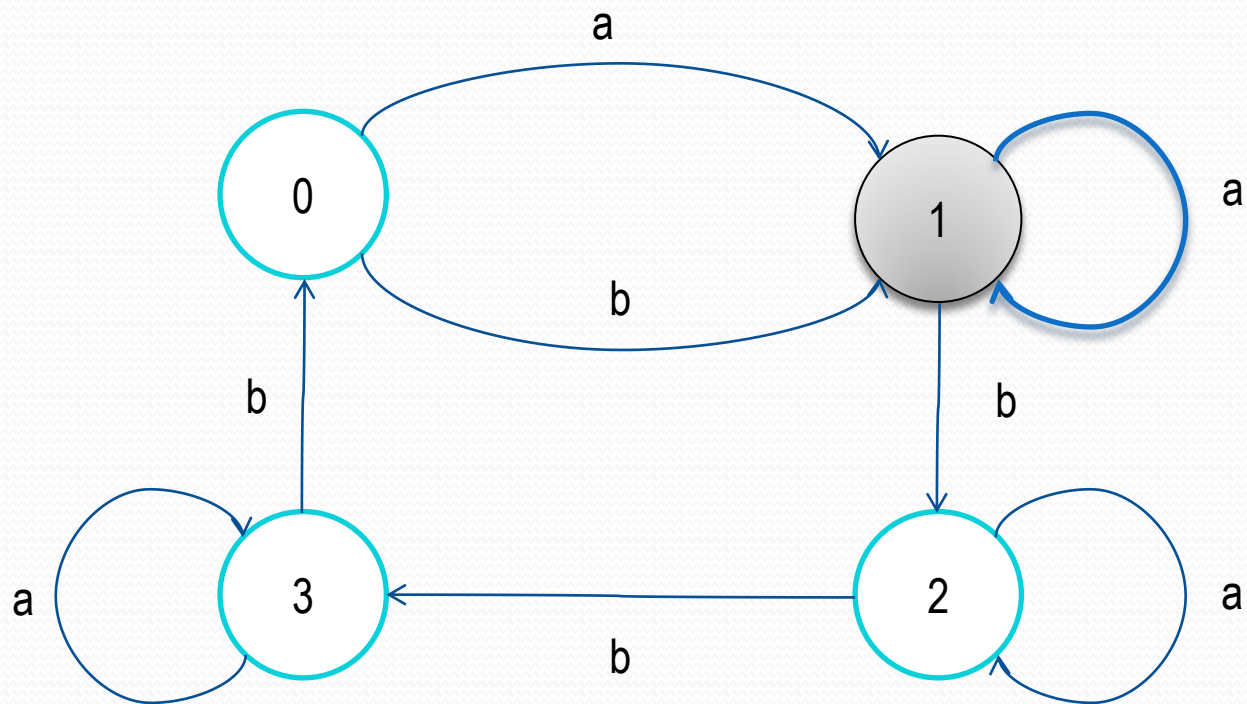
Šį automatą sinchronizuojantis žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



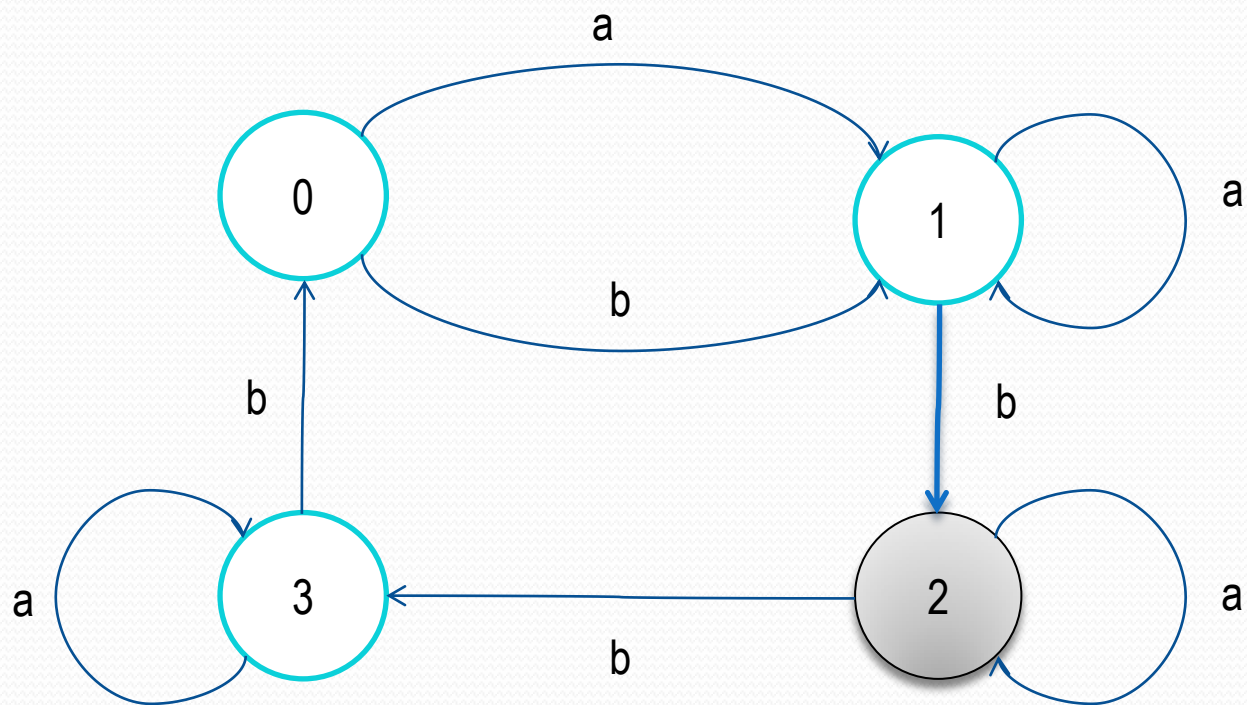
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



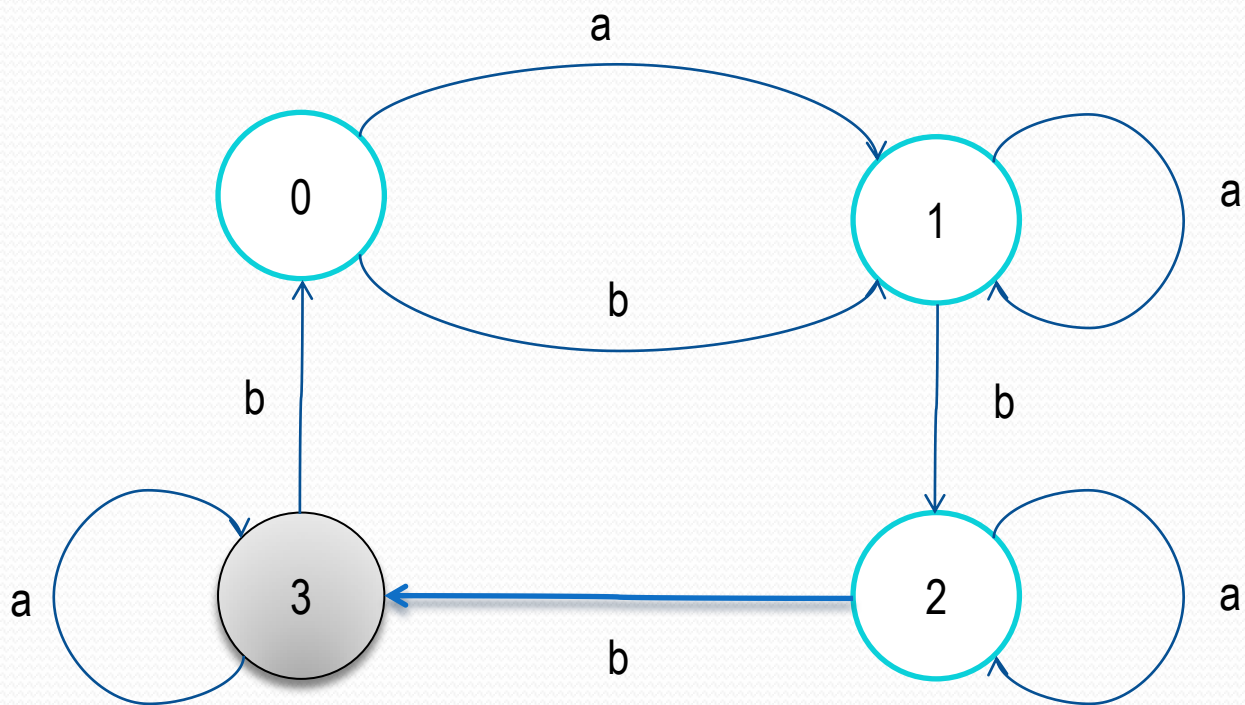
Šį automatą sinchronizuojančis žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



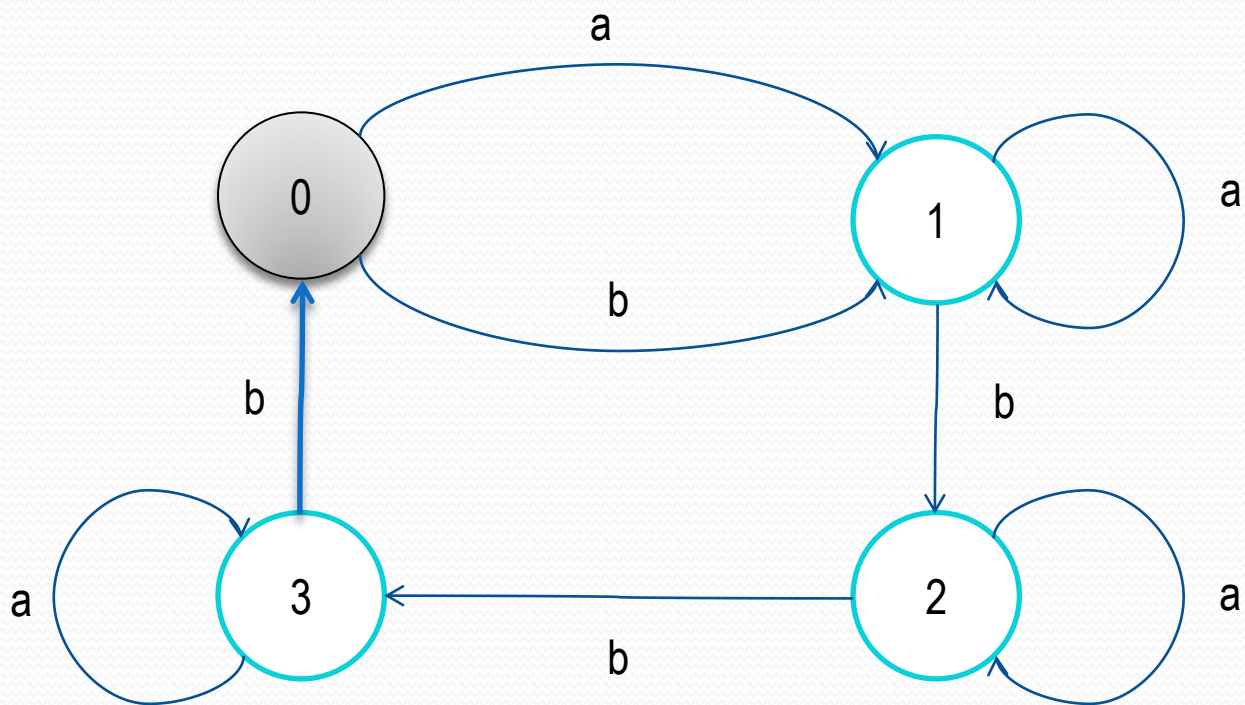
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



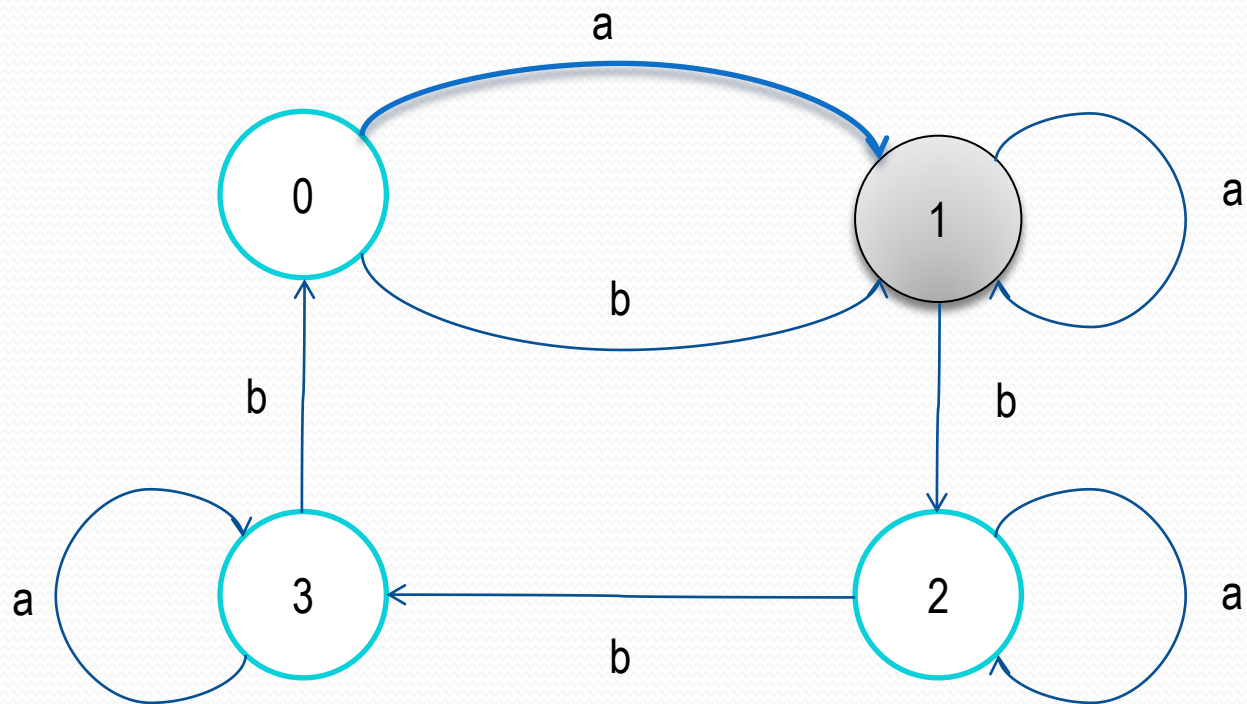
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



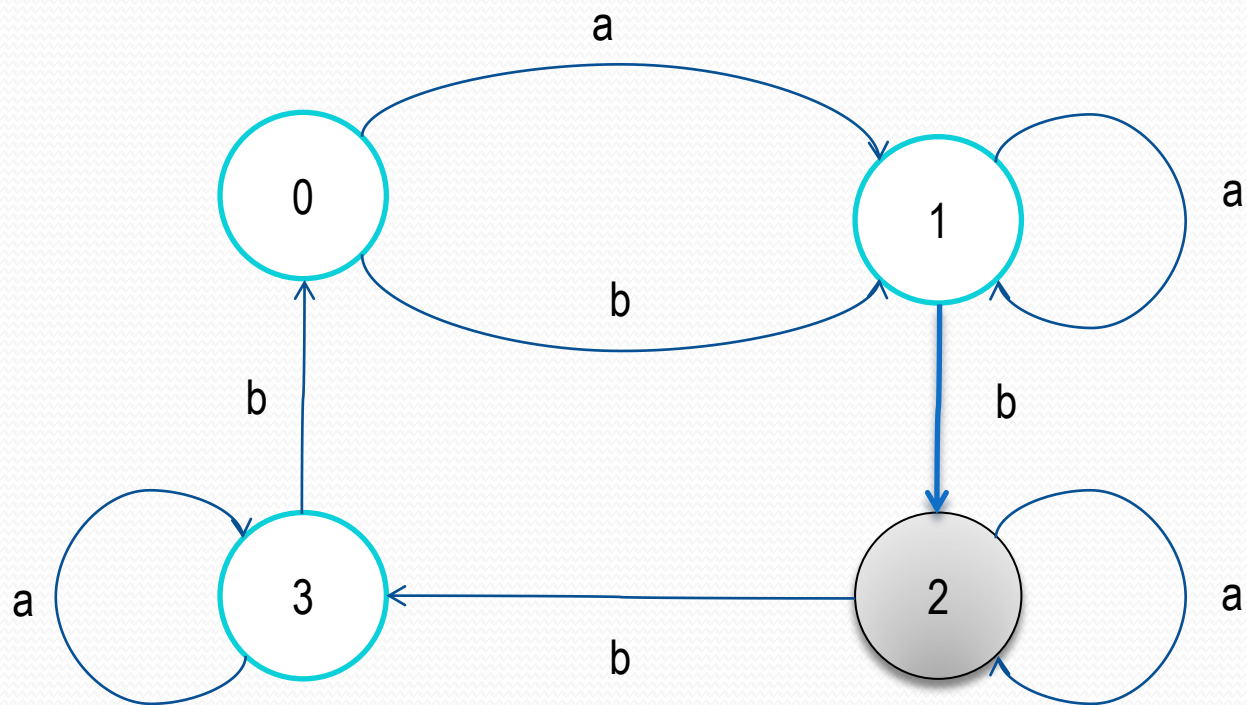
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



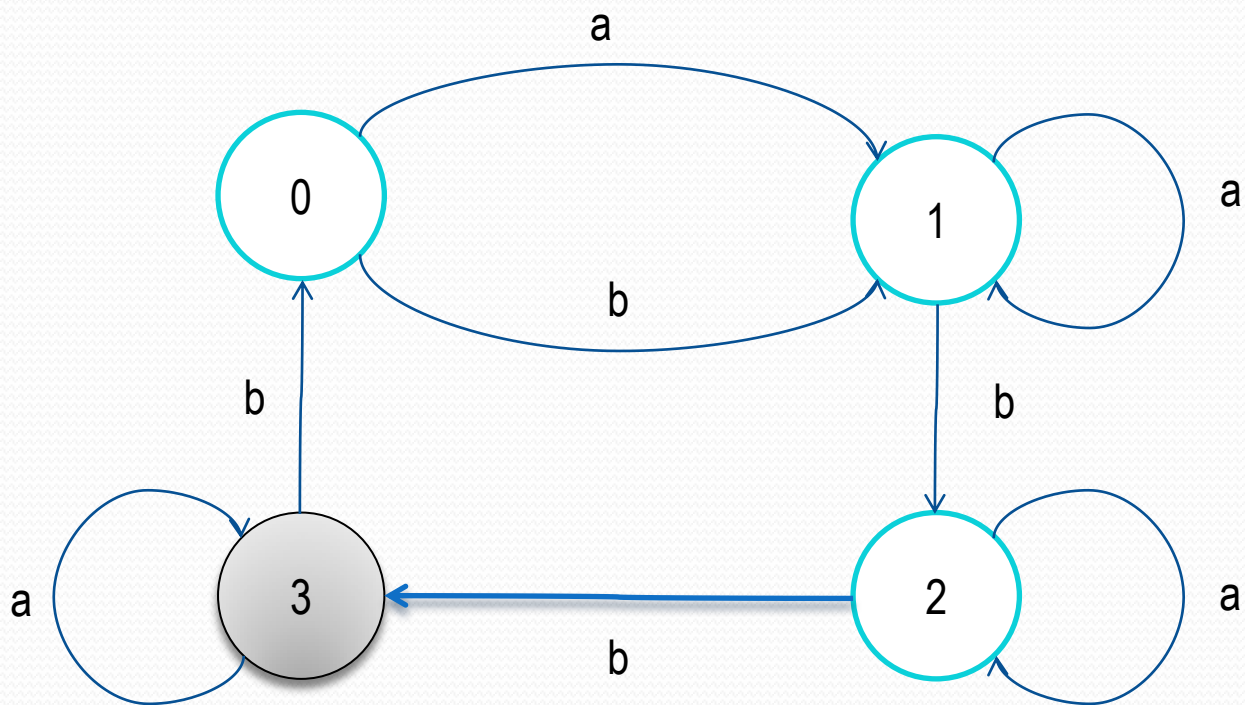
Šį automatą sinchronizuojantis žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



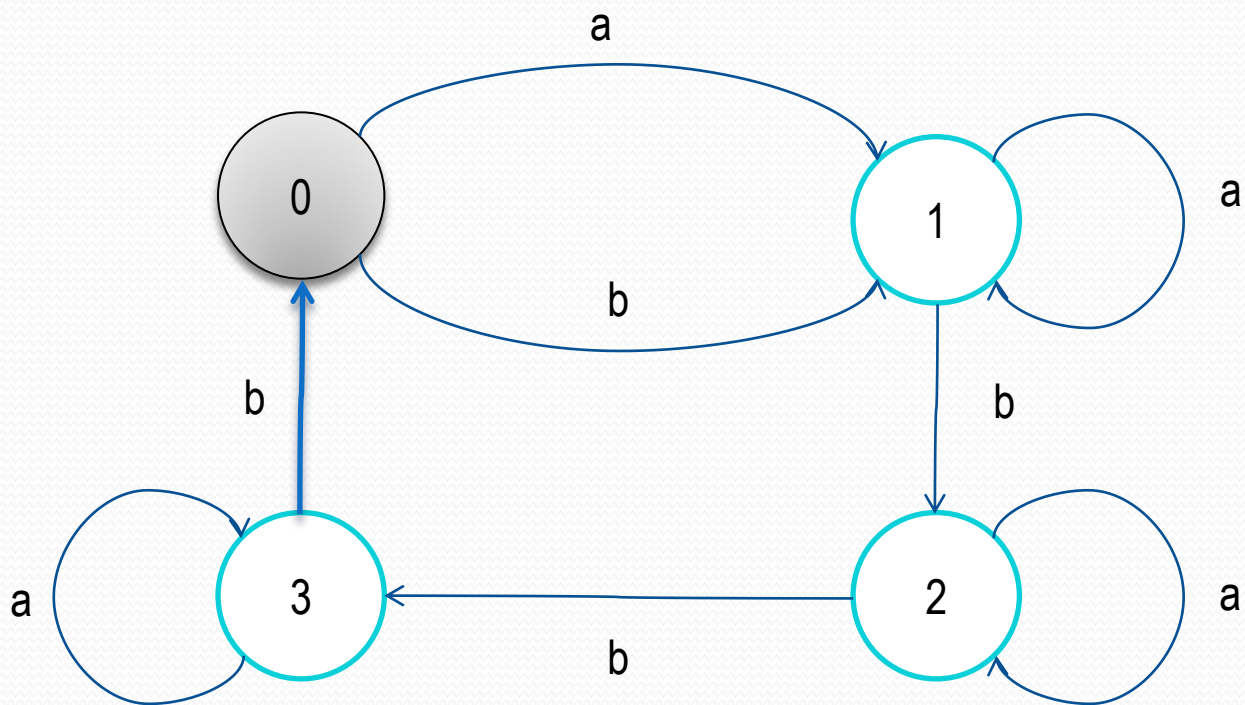
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



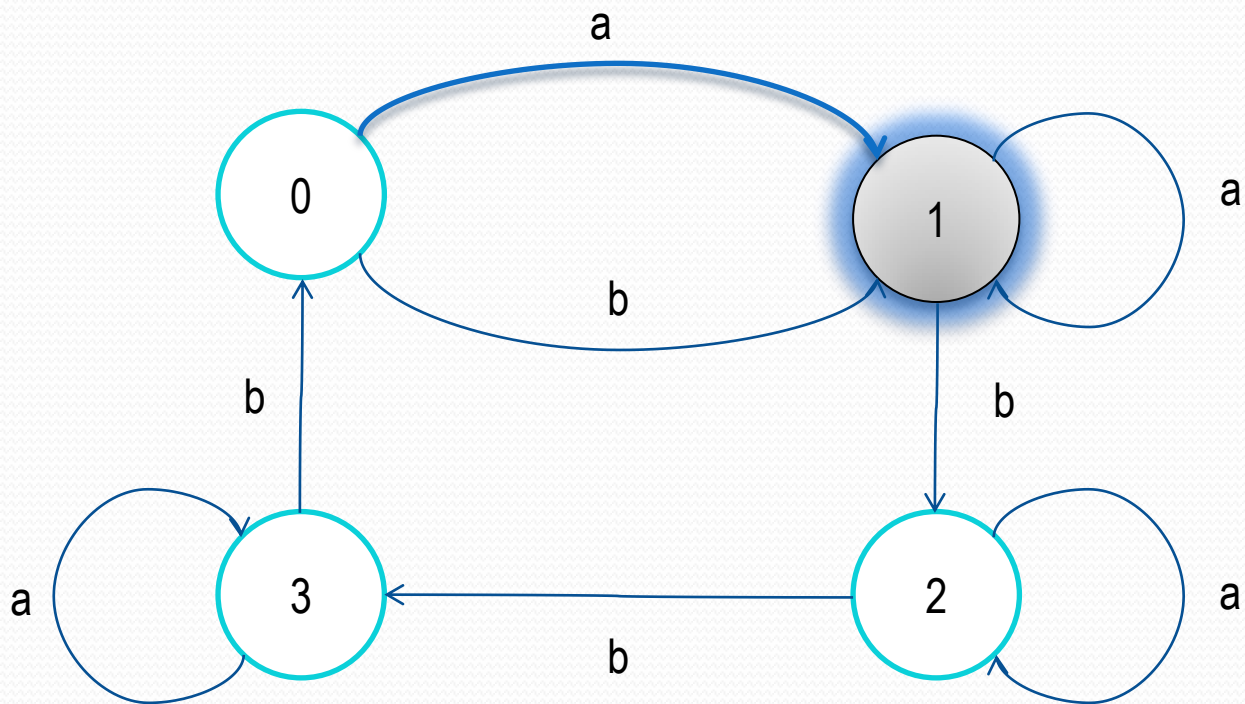
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



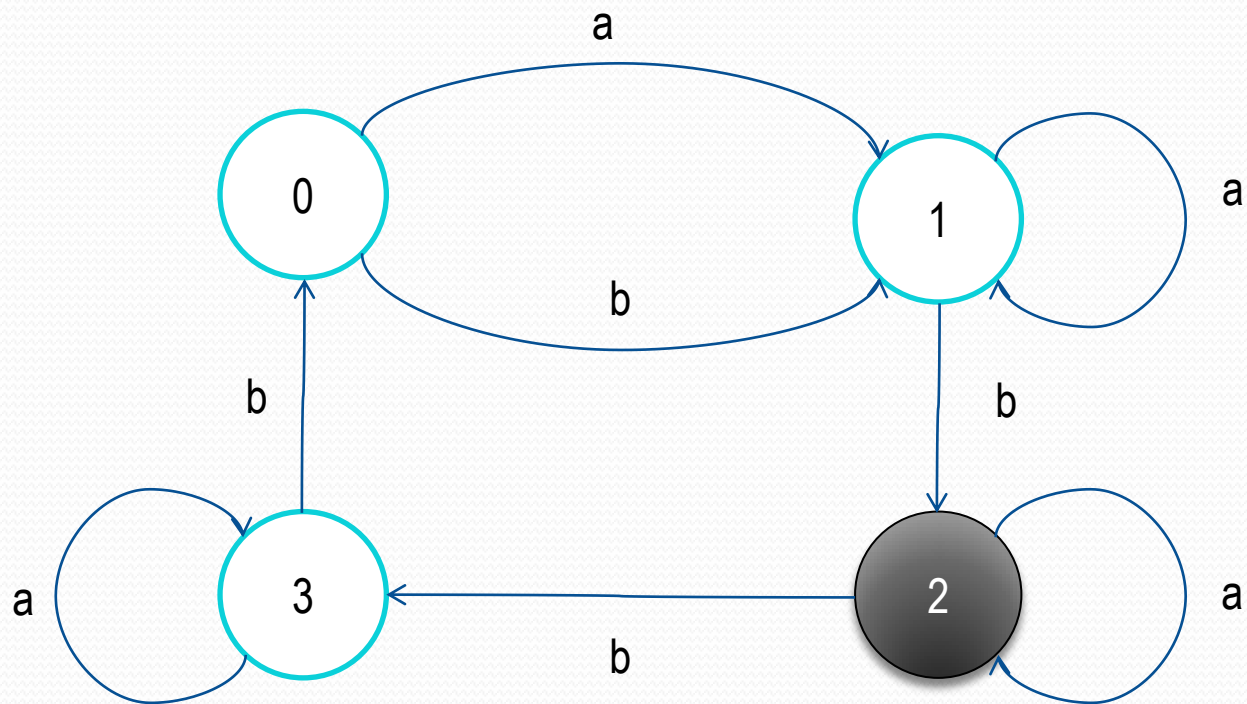
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



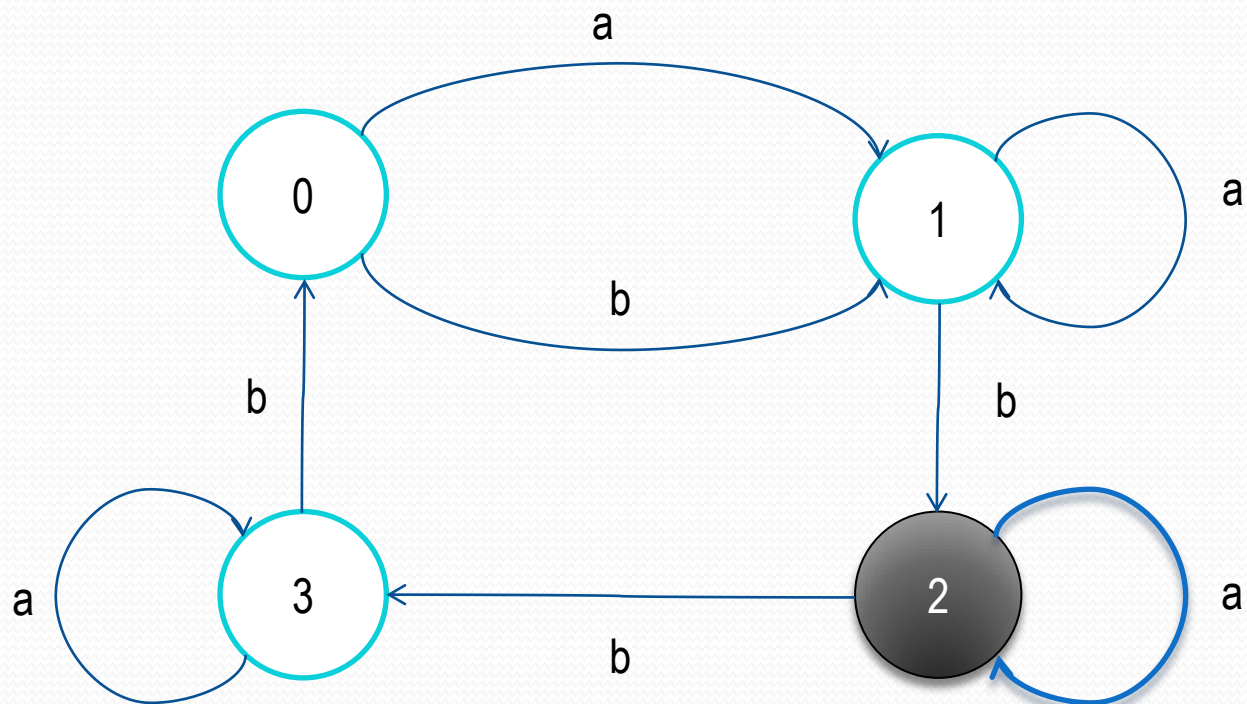
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



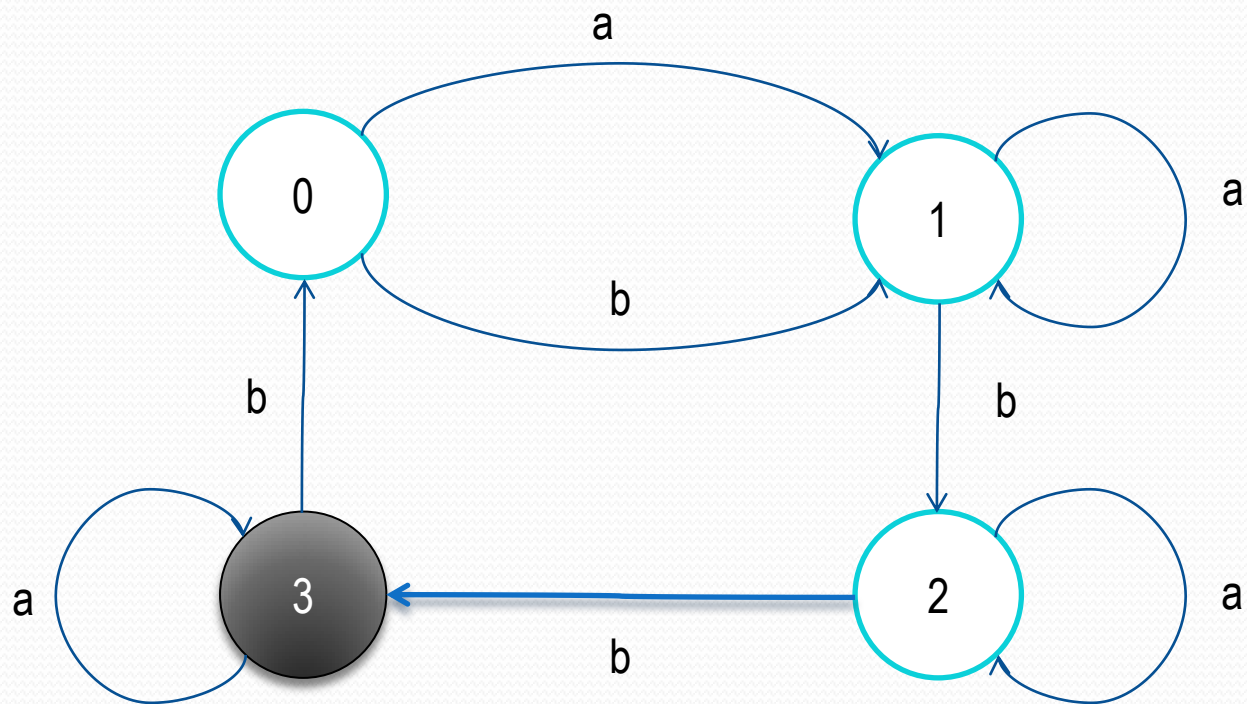
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



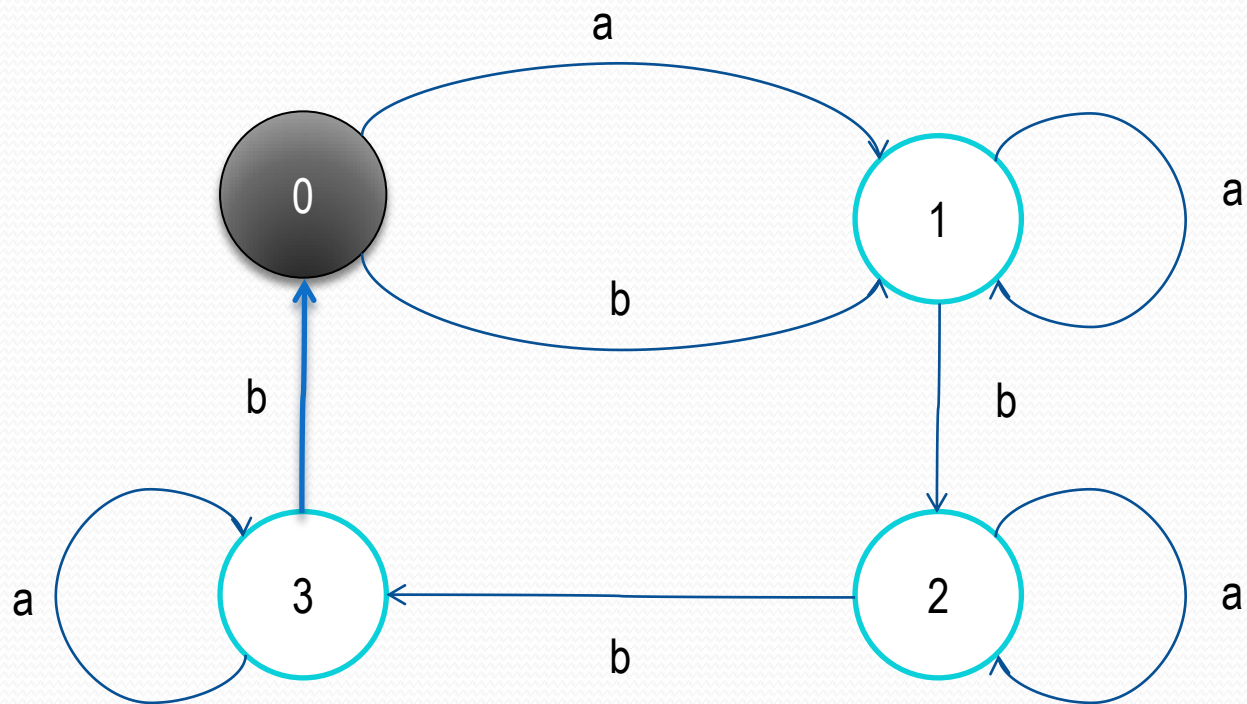
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



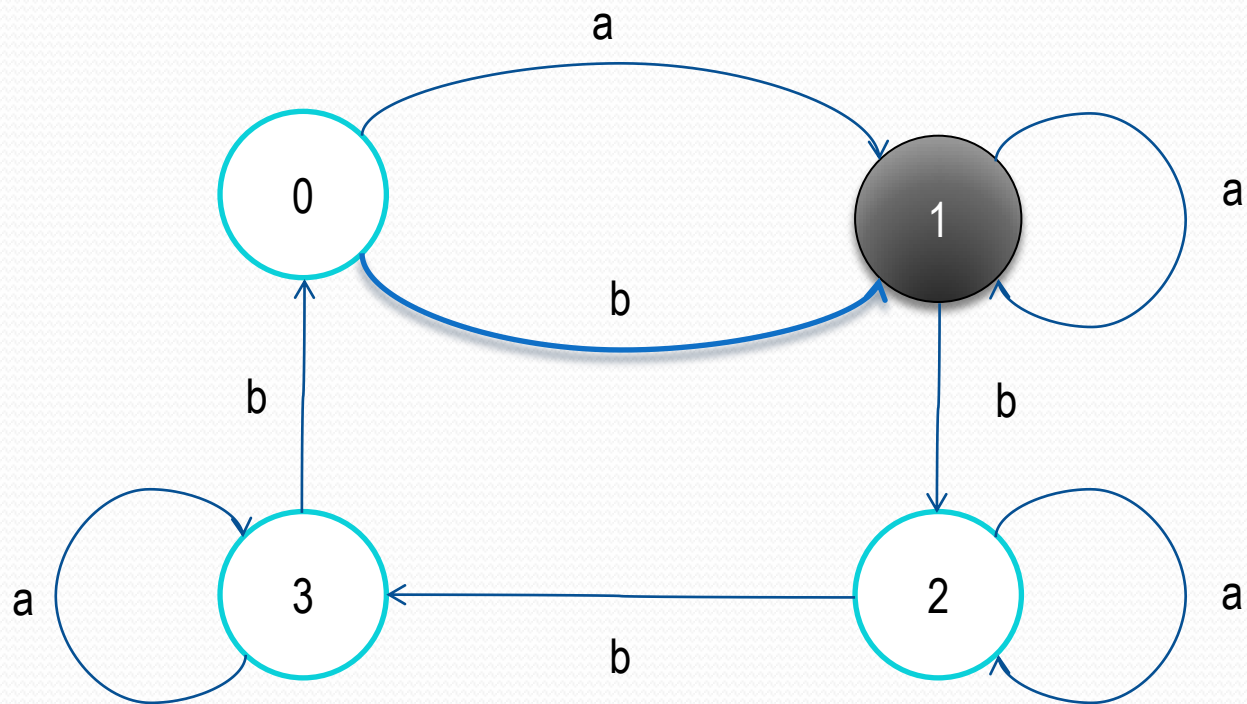
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



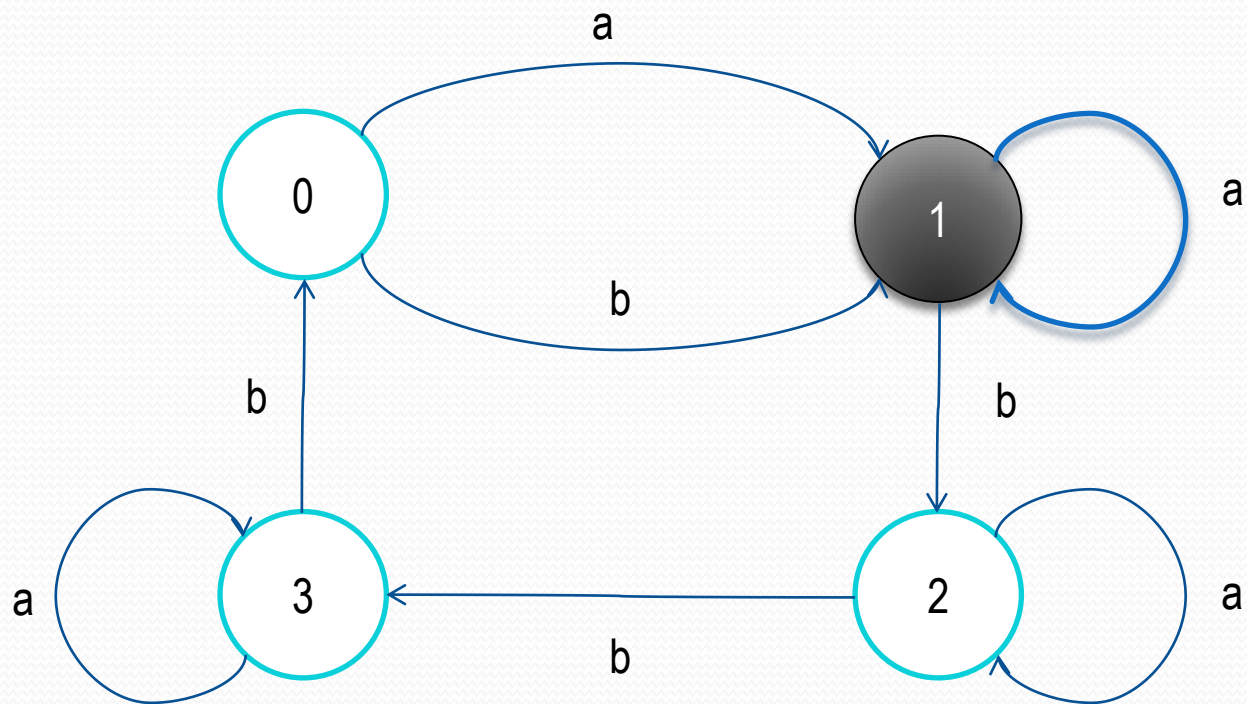
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



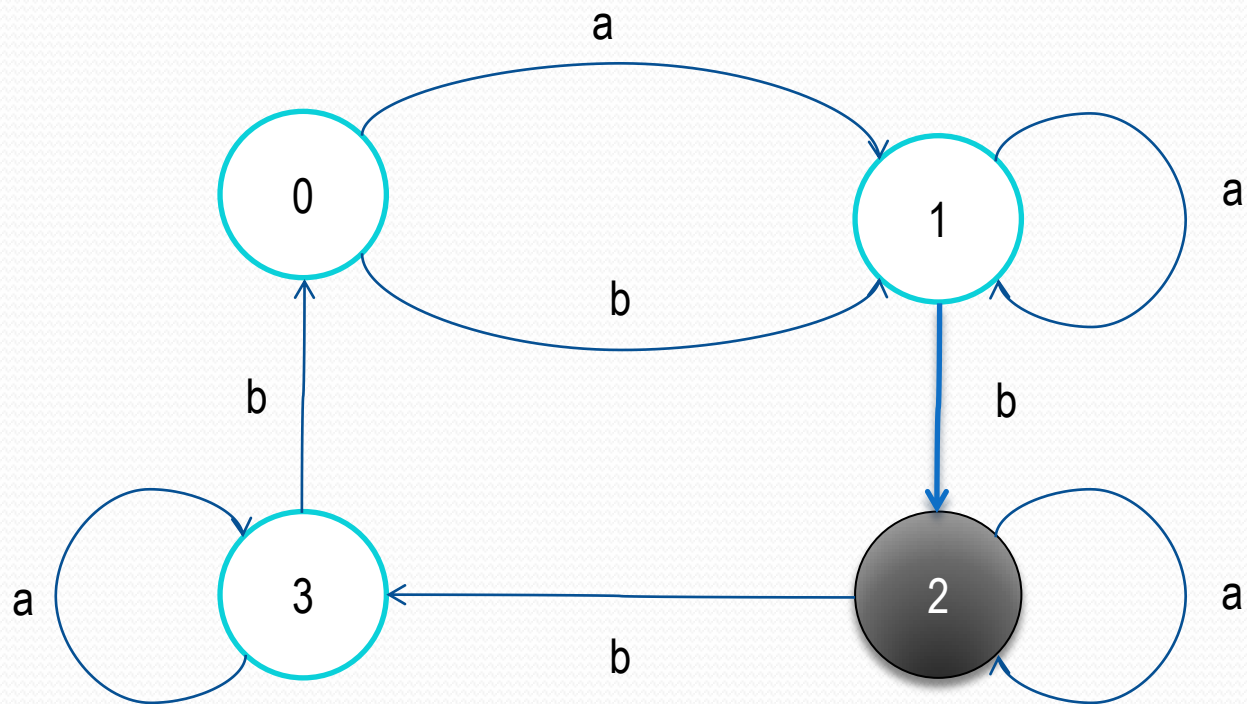
Šį automatą sinchronizuojančis žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



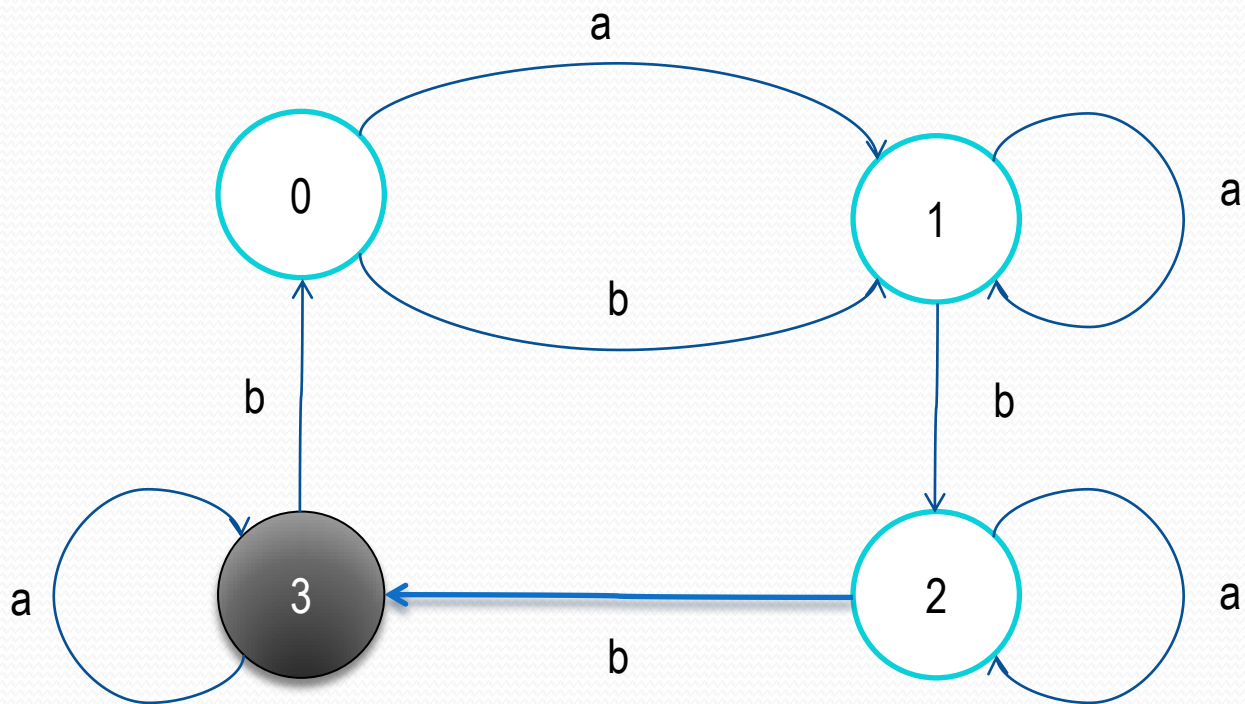
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



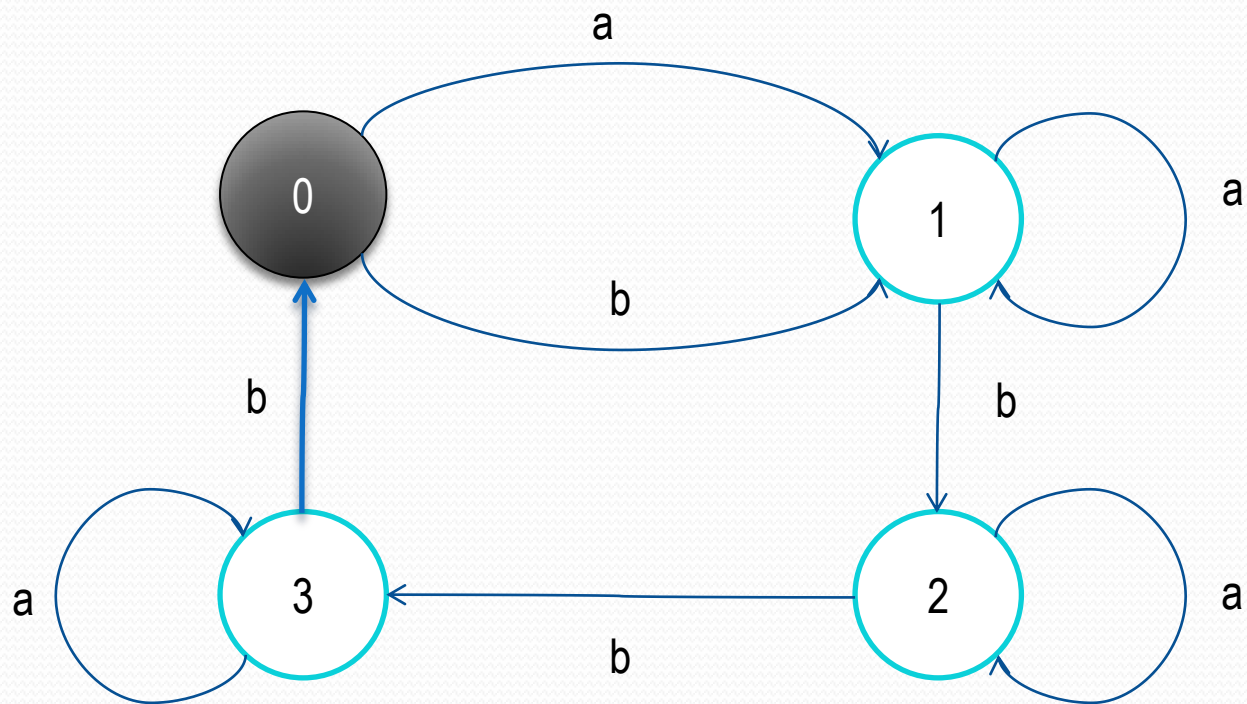
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



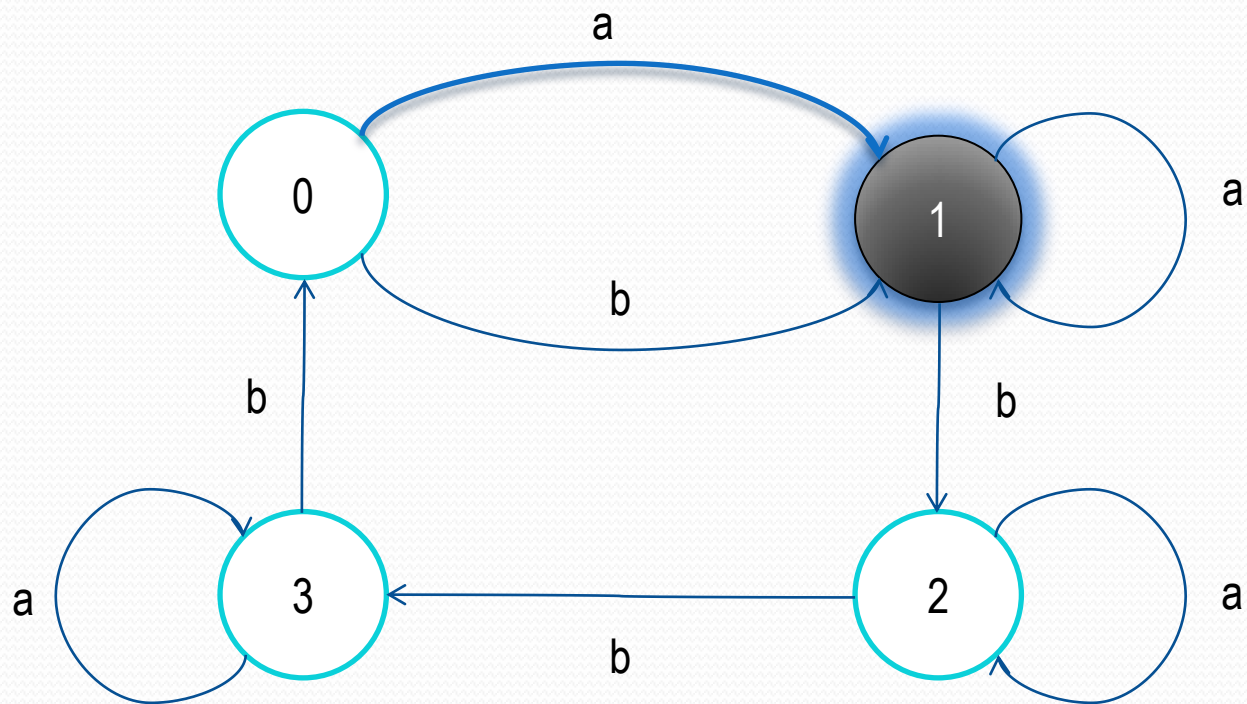
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



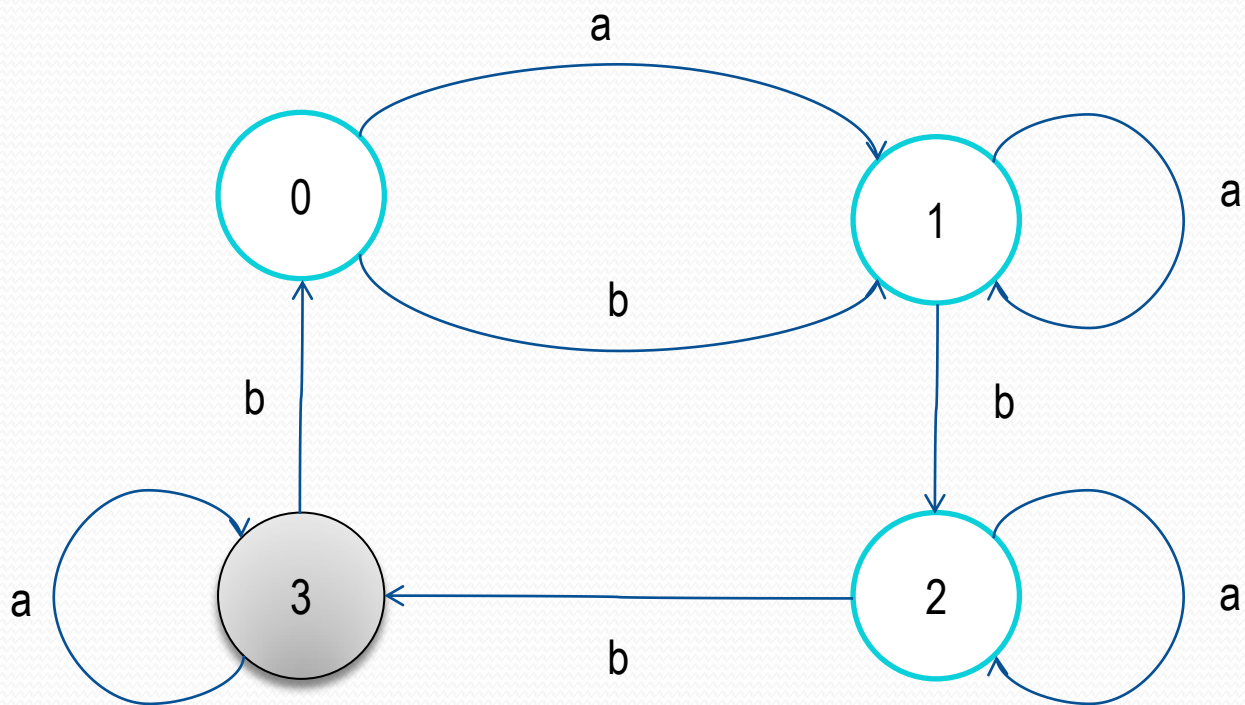
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



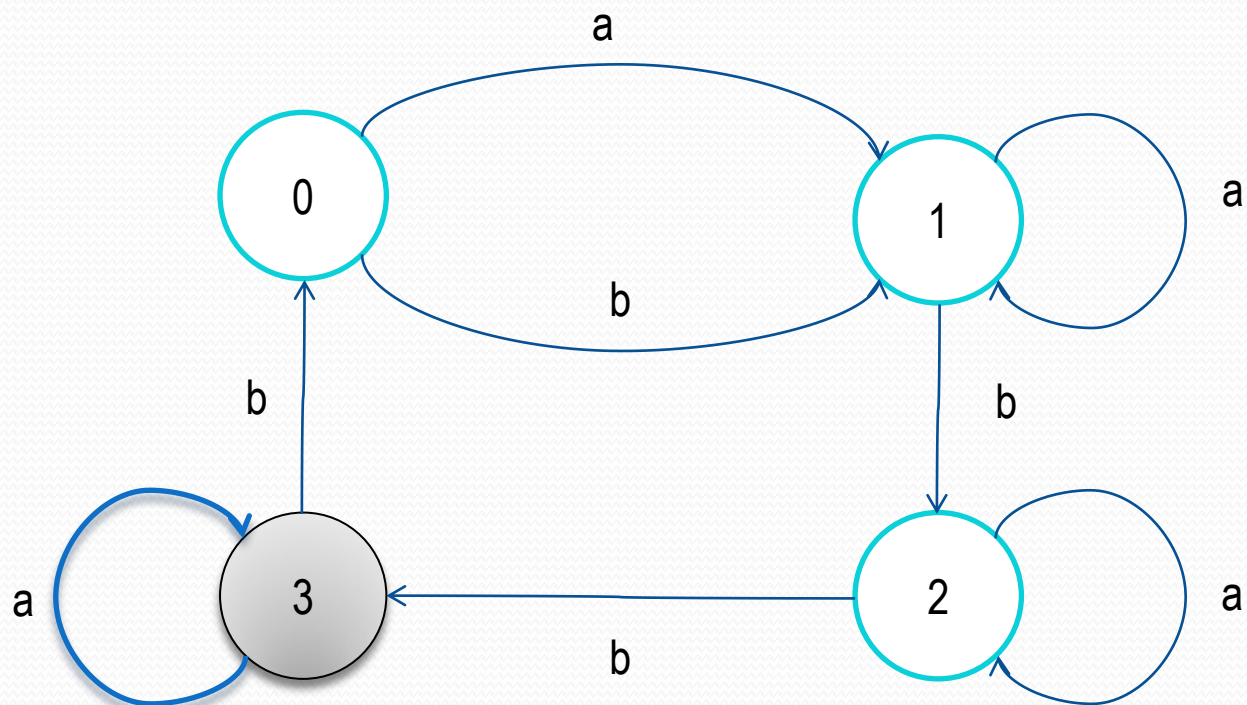
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



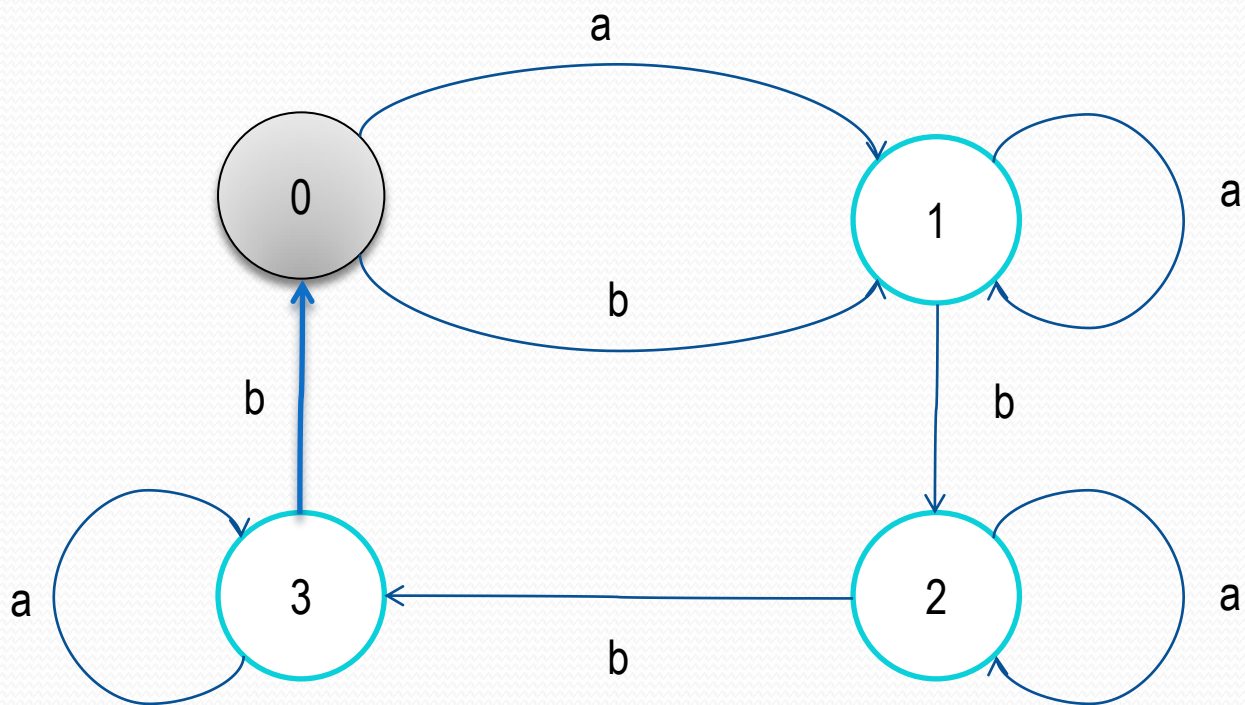
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



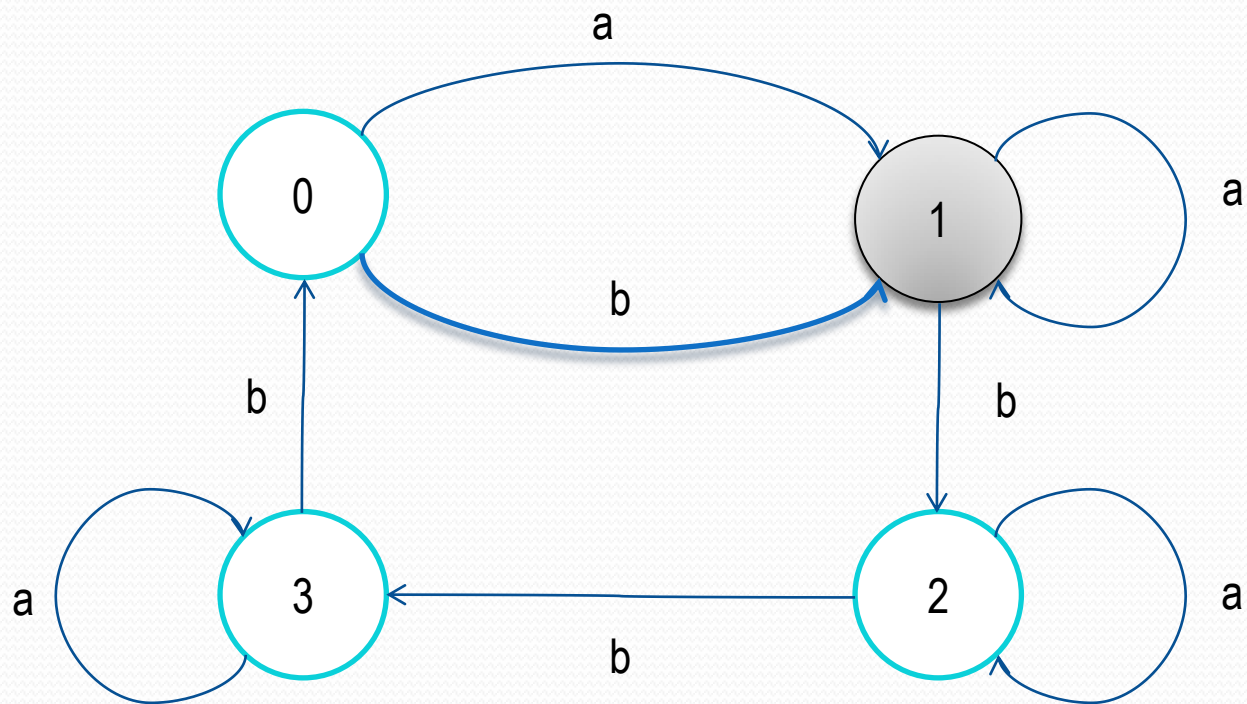
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



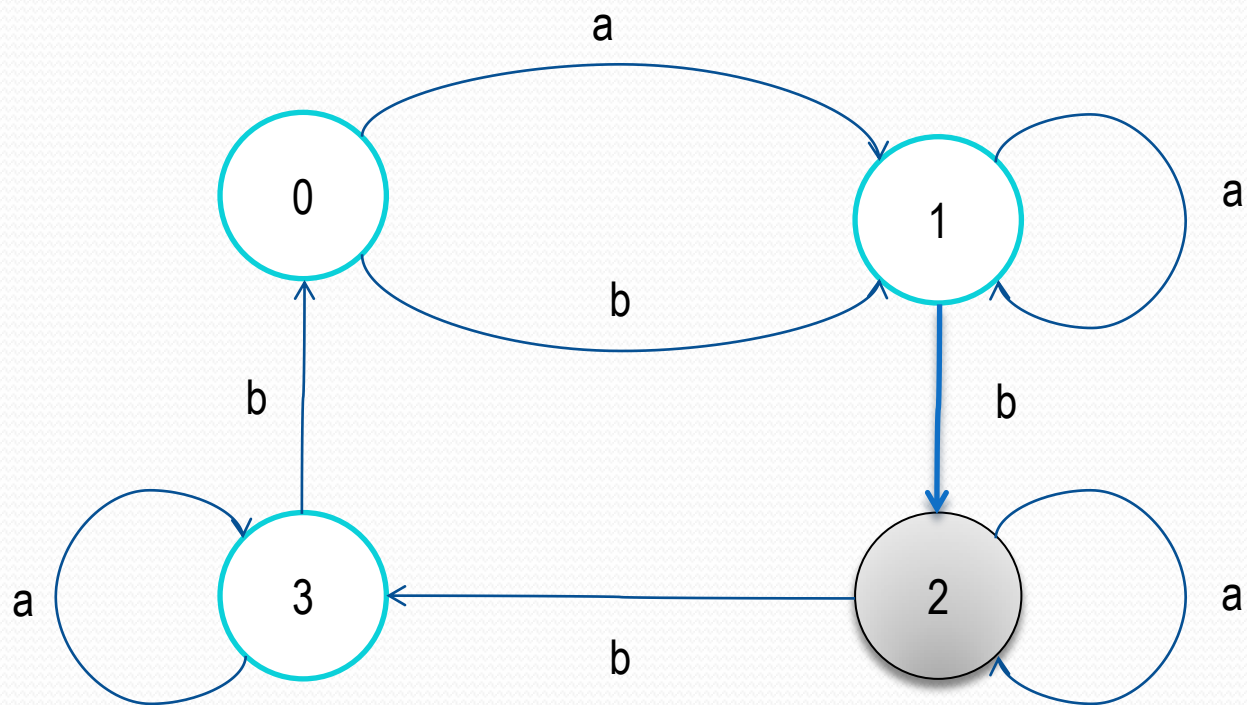
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



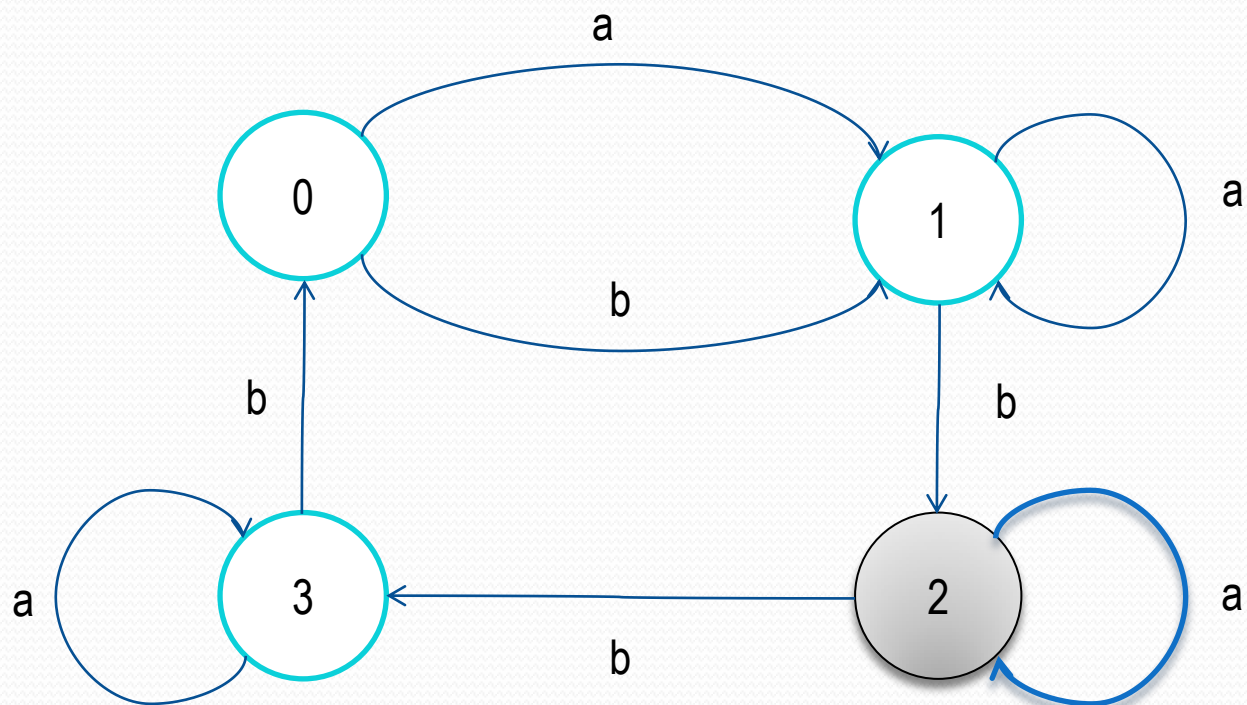
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



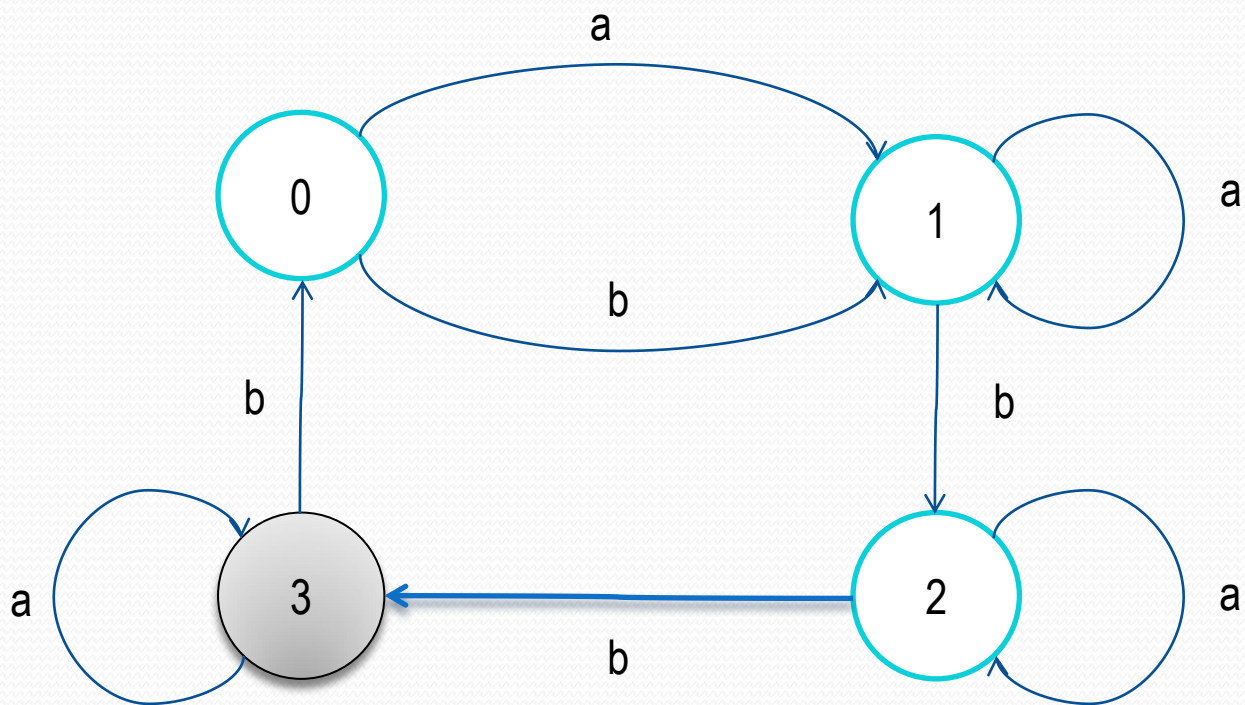
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



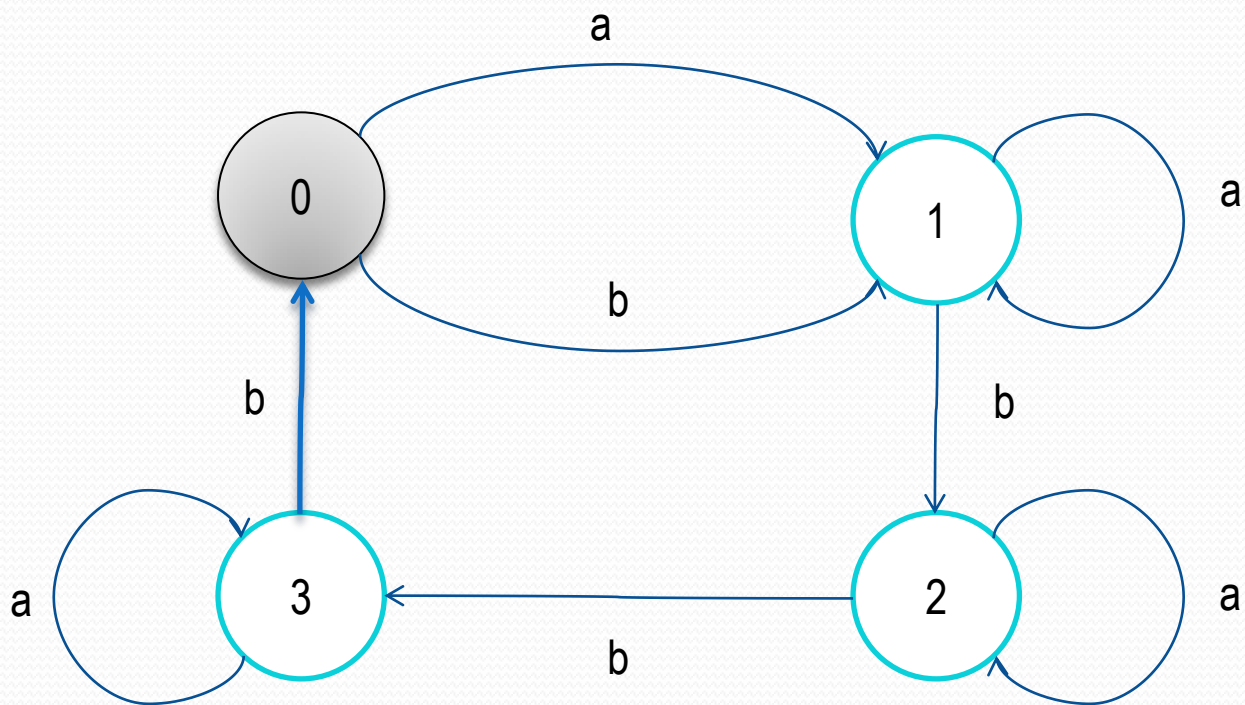
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



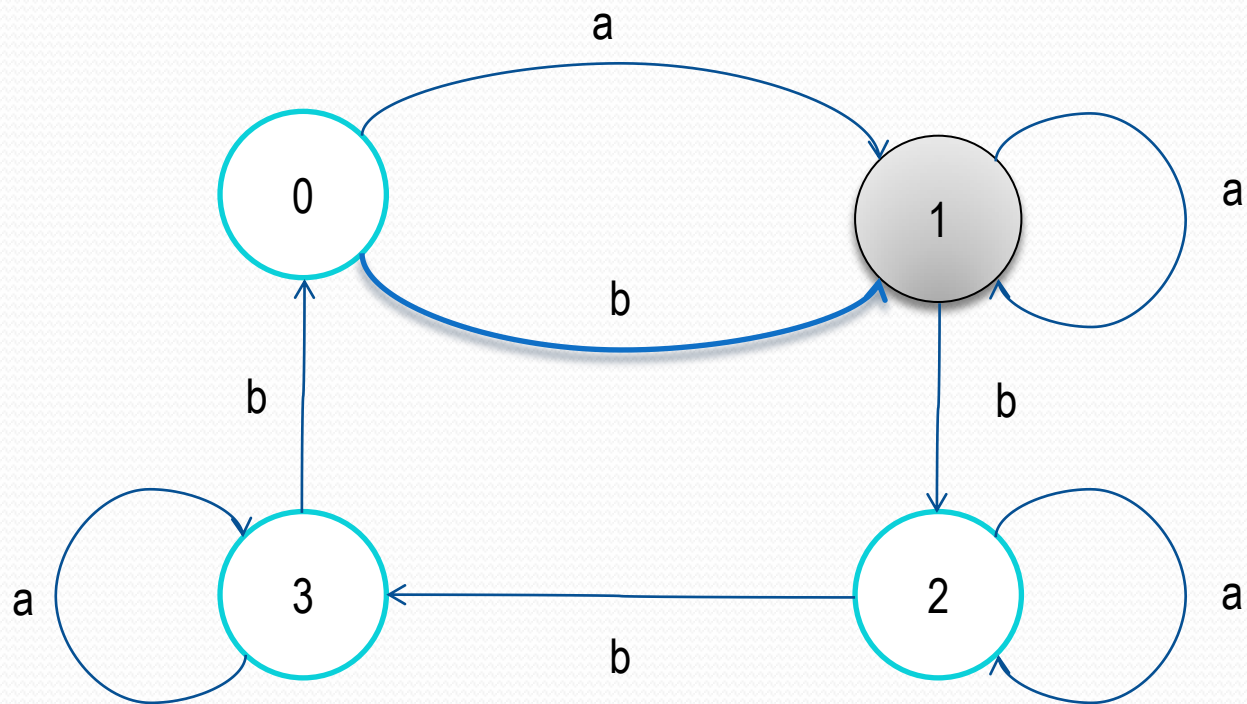
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



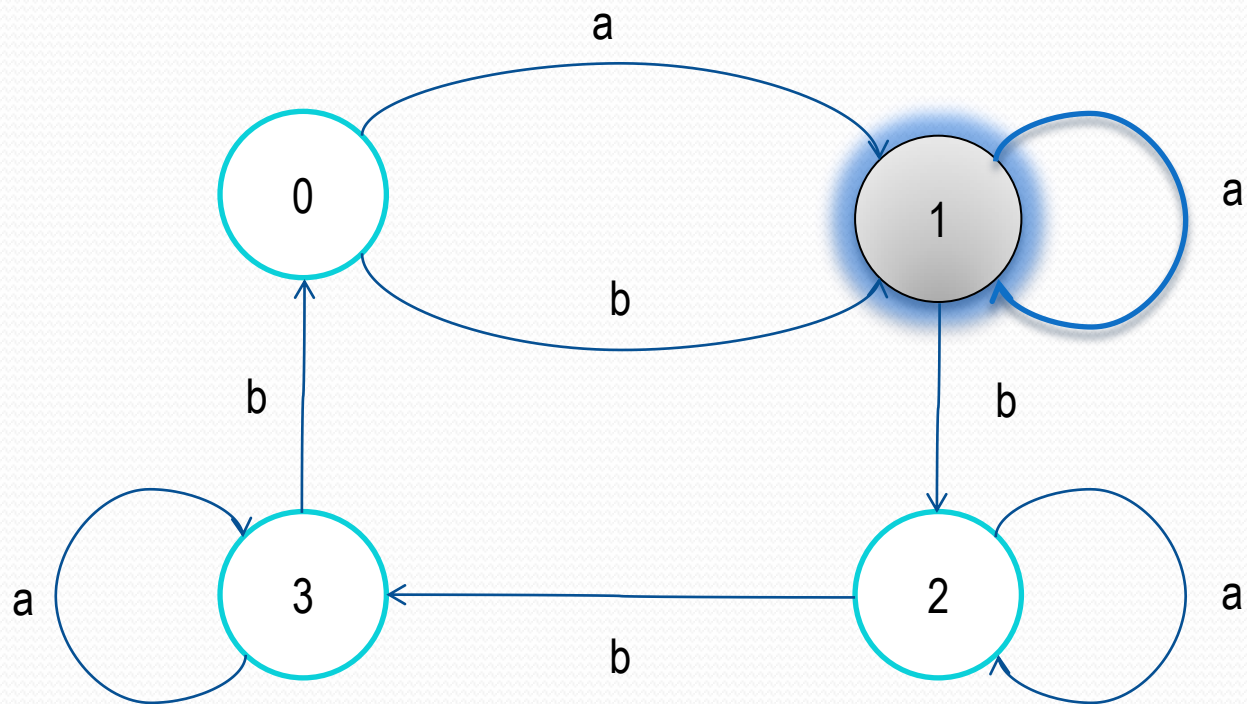
Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato pavyzdys



Šį automatą sinchronizuojančias žodis yra *abbbabbba*:
jis automatą perveda į būseną 1.

Sinchronizuojančio automato sąvoką pirmasis pateikė slovakių matematikas **Jan Černý** 1964 metais parašytame straipsnyje.

Jau iš pat pradžių sinchronizavimo idėja tapo populiaria: nesvarbu kokioje būsenoje yra įranga mes galime ją pervesti į pradinę, iš anksto žinomą būseną: įsivaizduokime palydovą, kuris skrieja aplink Mėnulį ir kurio mes Žemėje negalime kontroliuoti, kai jis „kitoje“ Mėnulio pusėje (Černý pavyzdys).

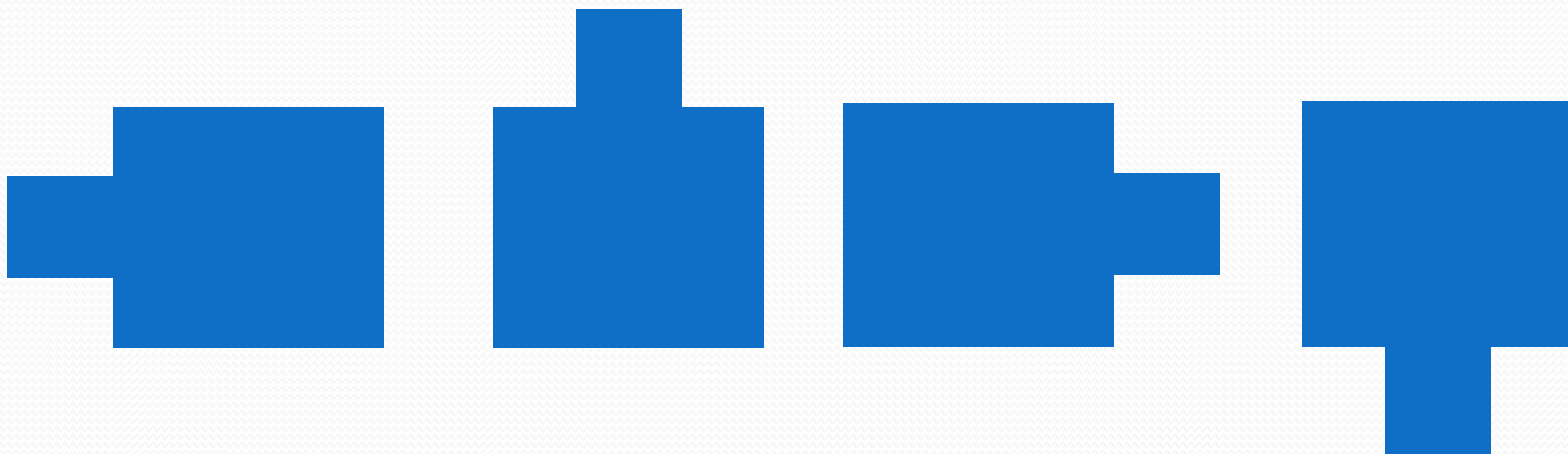
Nuo XX amžiaus 9-ojo dešimtmečio šią idėją pradėta įgyvendinti automatizuojant įvairius gamybos procesus, kuriant įvairius pramoninius robotus.

Pateiksime pavyzdį.

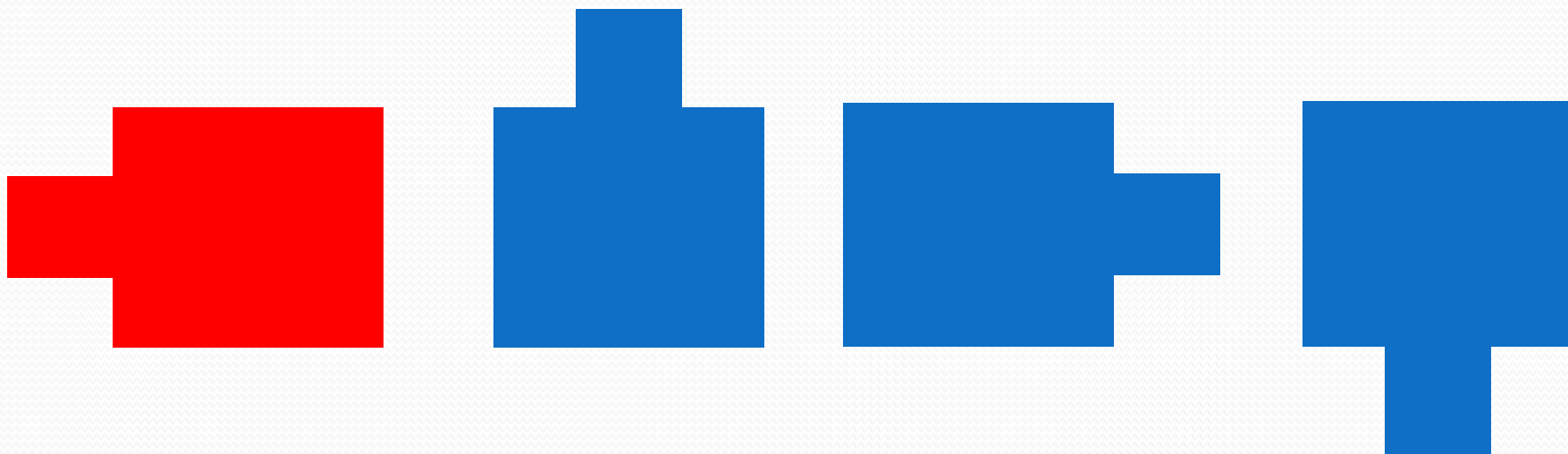
Nagrinėkime įrenginį, kuriam sukonstruoti naudojama detalė-sensorius



Detalė konvejeriye gali būti vienoje iš keturių padėčių :



Detalė konvejeriye gali būti vienoje iš keturių padėčių :

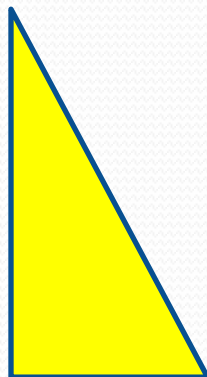
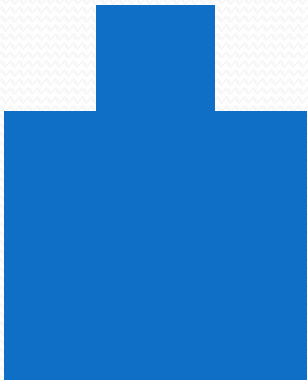


Ir tik pirmoji padėtis tinkama montavimui įrenginyje.

Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

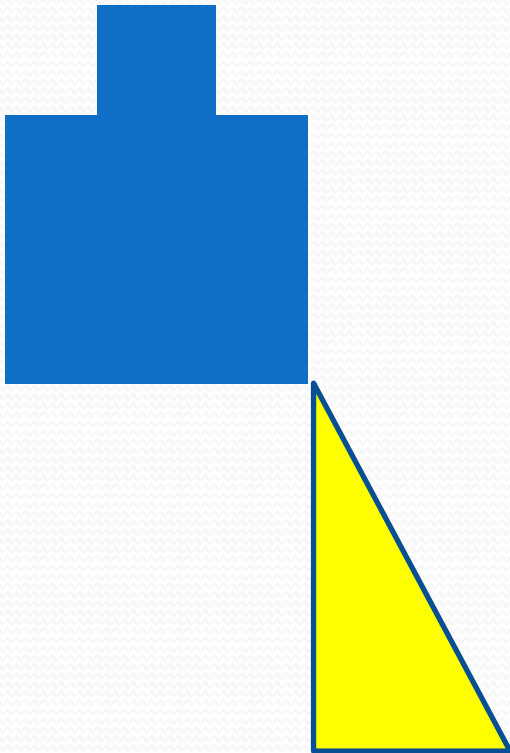
Aukštoji kliūtis gali pasiekti judančios detalės dešinįjį apatinį kampą ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

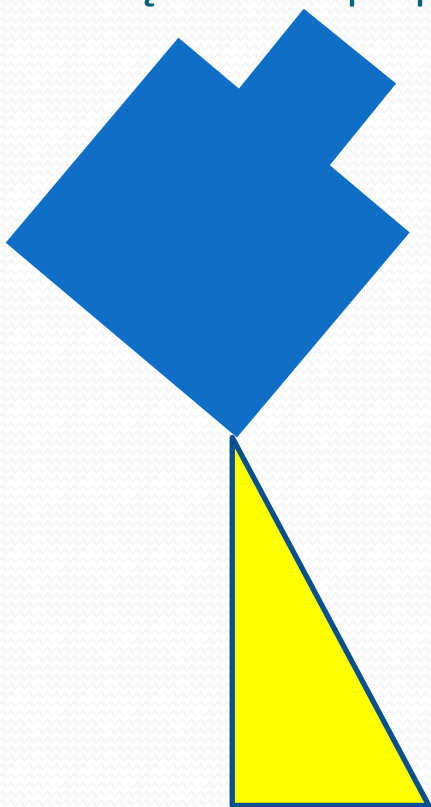
Aukštoji kliūtis gali pasiekti judančios detalės dešinįjį apatinį kampą ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

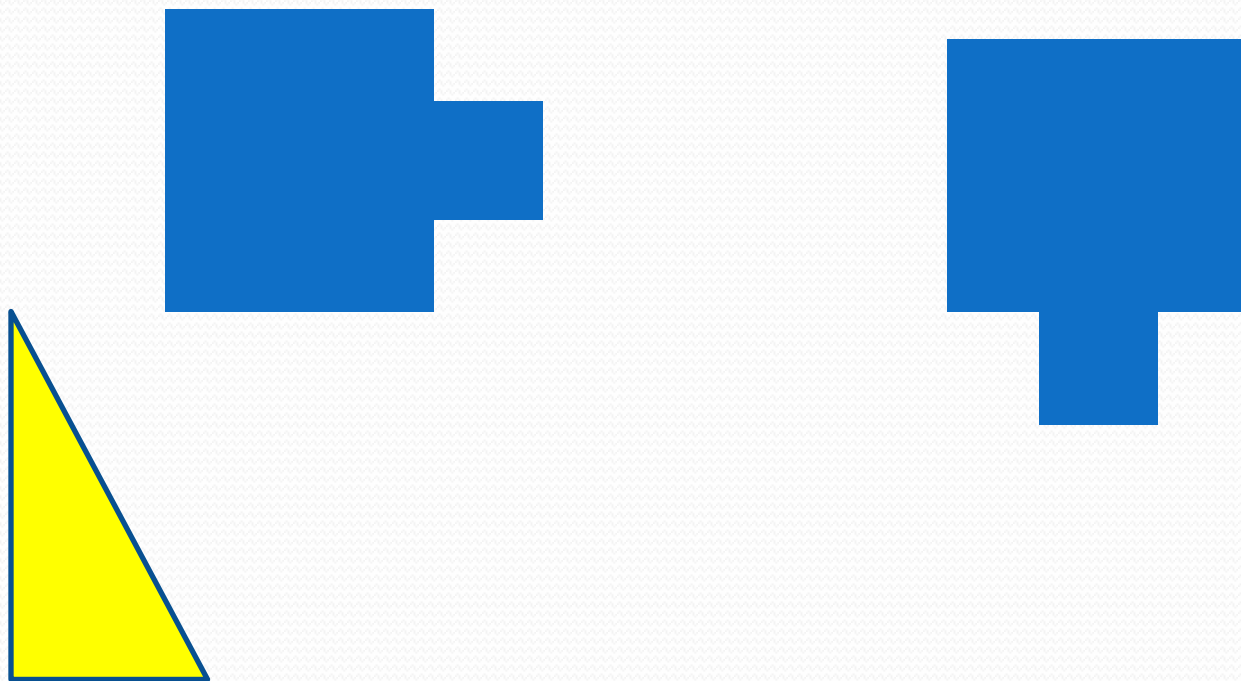
Aukštoji kliūtis gali pasiekti judančios detalės dešinįjį apatinį kampą ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

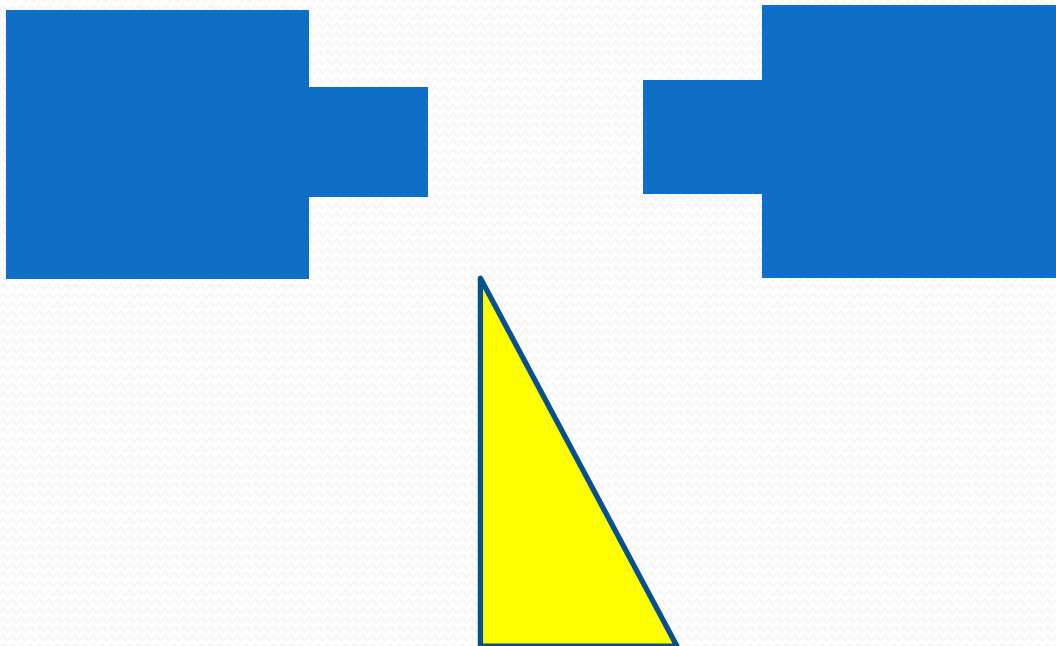
Aukštoji kliūtis gali pasiekti judančios detalės dešinįjį apatinį kampą ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

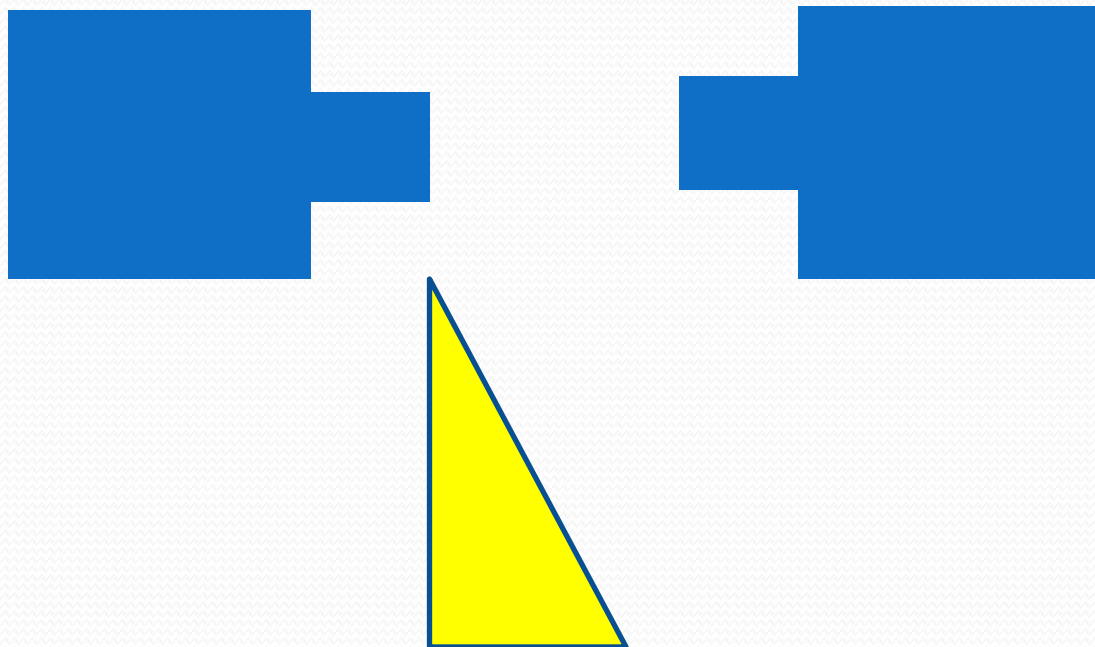
Aukštoji kliūtis gali pasiekti judančios detalės dešinįjį apatinį kampą ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

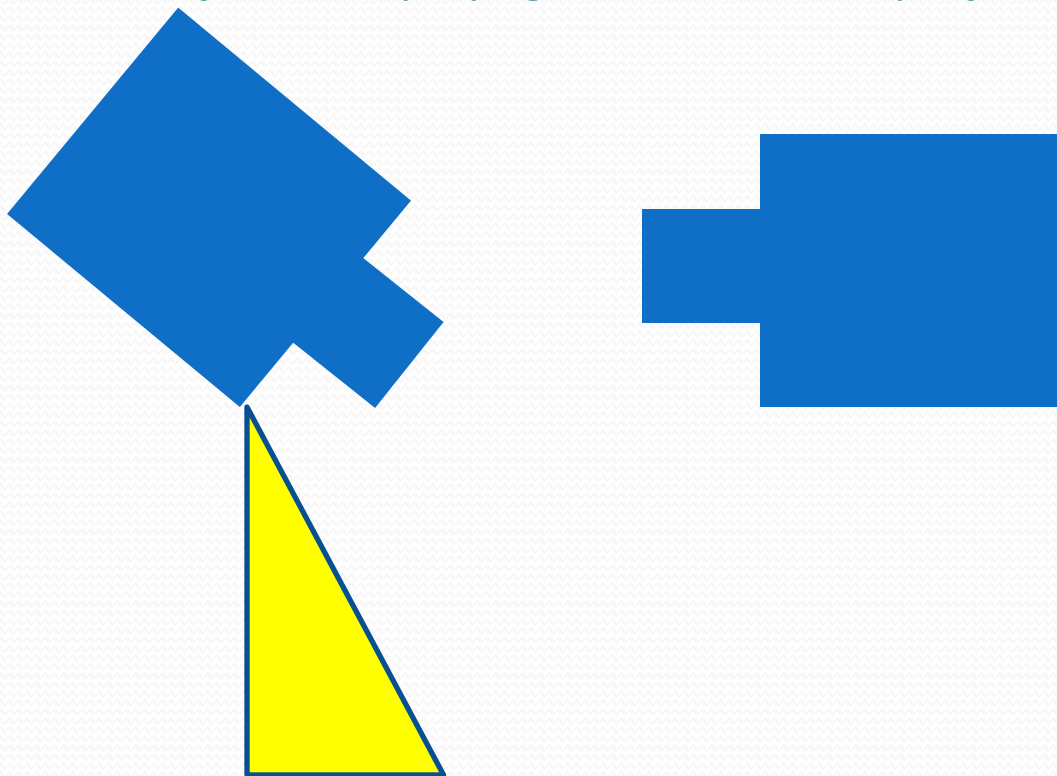
Aukštoji kliūtis gali pasiekti judančios detalės dešiniąją apatinę kampą ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

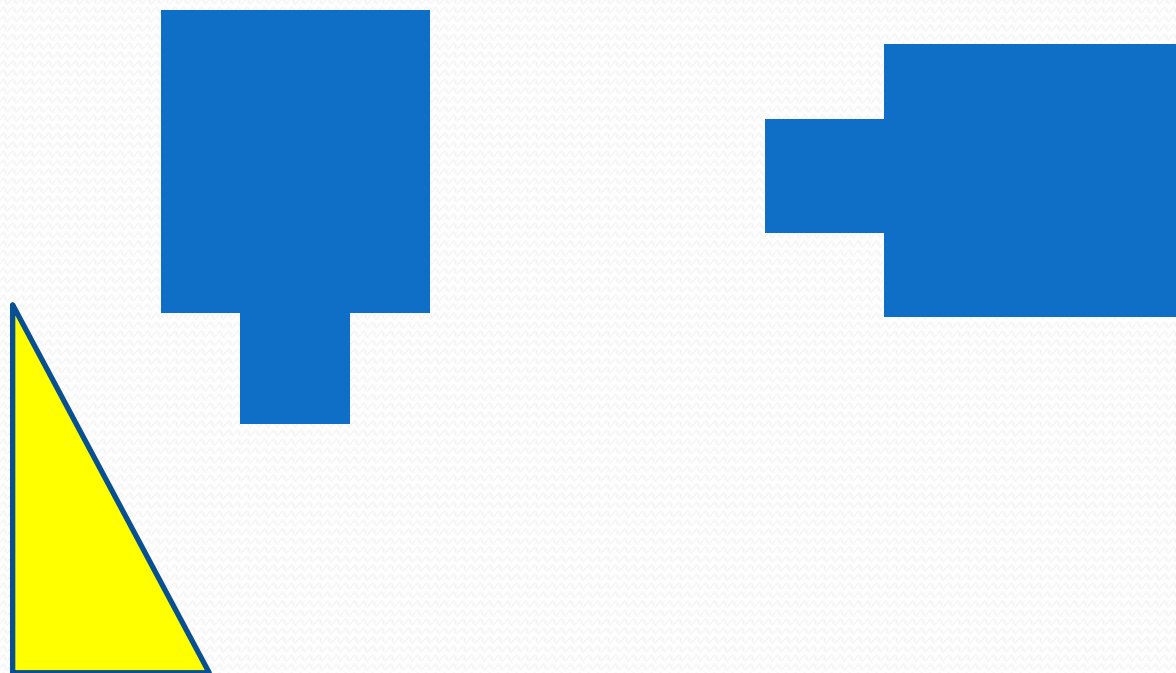
Aukštoji kliūtis gali pasiekti judančios detalės dešinįjį apatinį kampą ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

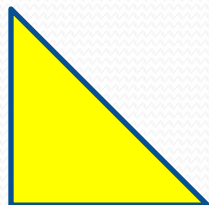
Aukštoji kliūtis gali pasiekti judančios detalės dešiniąjį apatinį kampą ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

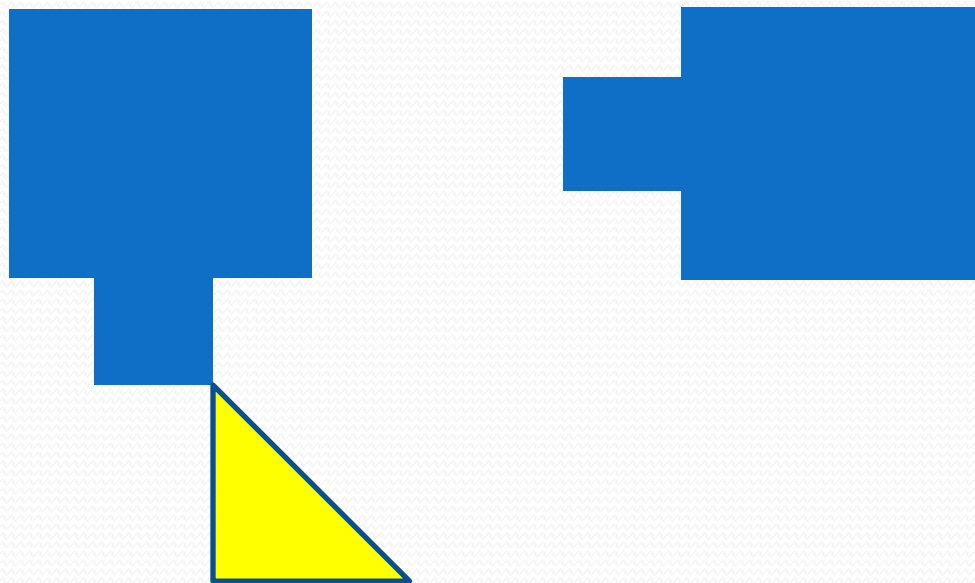
Žemoji kliūtis gali pasiekti judančios detalės dešinįjį apatinį kampą (jeigu pasiekia) ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

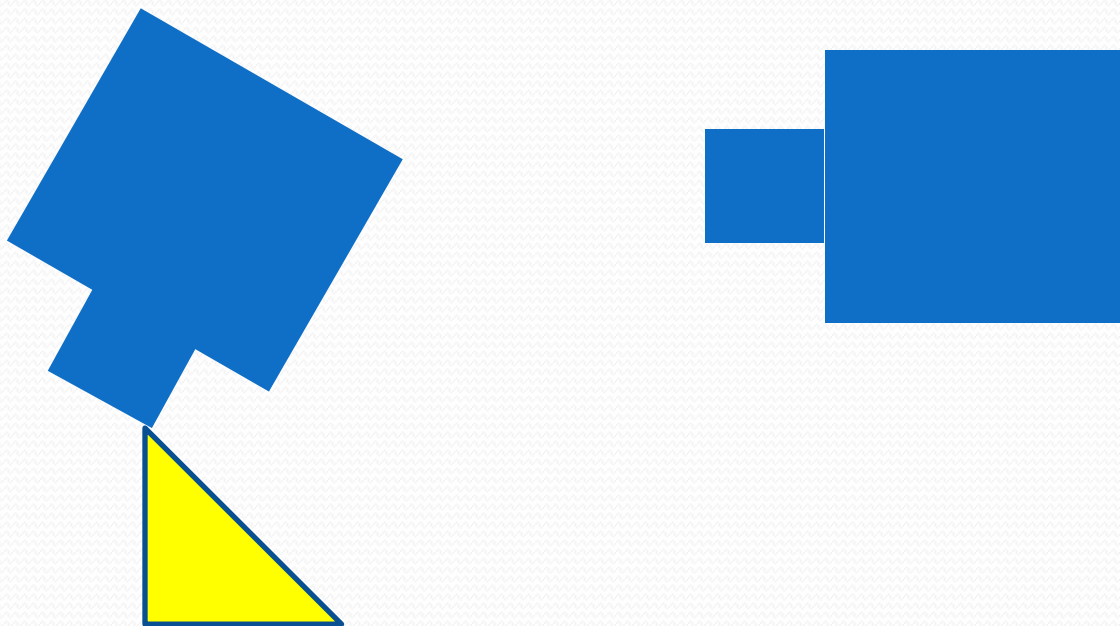
Žemoji kliūtis gali pasiekti judančios detalės dešinįjį apatinį kampą (jeigu pasiekia) ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

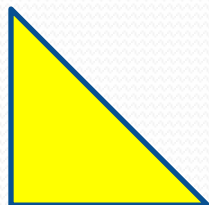
Žemoji kliūtis gali pasiekti judančios detalės dešinįjį apatinį kampą (jeigu pasiekia) ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

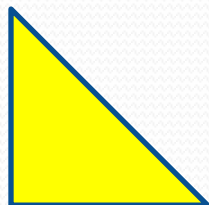
Žemoji kliūtis gali pasiekti judančios detalės dešinįjį apatinį kampą (jeigu pasiekia) ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

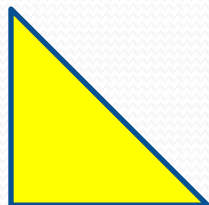
Žemoji kliūtis gali pasiekti judančios detalės dešinįjį apatinį kampą (jeigu pasiekia) ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



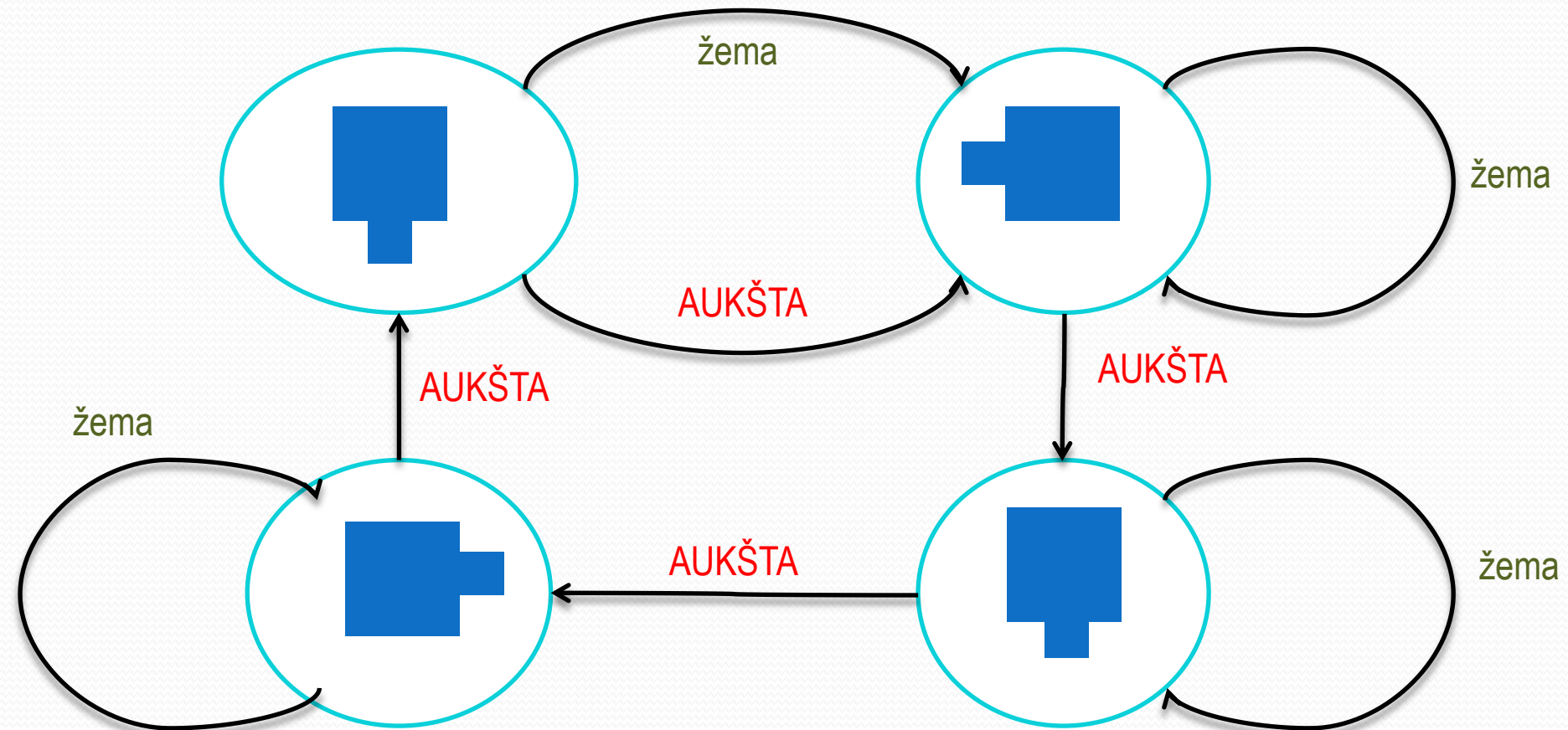
Konvejeris su jame nuosekliai sudėtomis detalėmis, esančiomis bet kurioje padėtyje, juda iš kairės į dešinę.

Savo kelyje detalės gali sutikti dviejų rūšių stacionarias kliūtis: aukštąją ir žemąją.

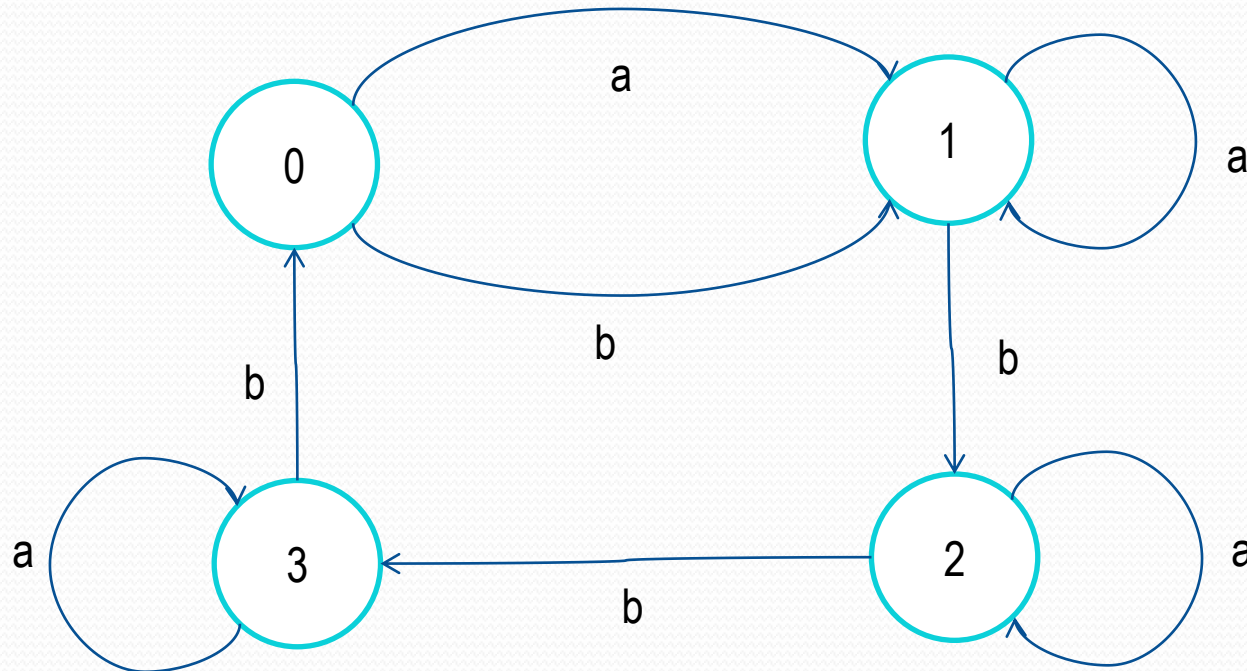
Žemoji kliūtis gali pasiekti judančios detalės dešinįjį apatinį kampą (jeigu pasiekia) ir pasukti detalę 90° kampu pagal laikrodžio rodyklę



Tokį „kliūčių ruožo“ įveikimą galime pavaizduoti diagrama



Palyginkime tai su aukščiau pateiktu automatu



Tai sinchronizuojantis automatas, kurio jį sinchronizuojantis žodis yra *abbbabbba*.

Todėl klūčių seka

žema-AUKŠTA-AUKŠTA-AUKŠTA-žema-AUKŠTA-AUKŠTA-AUKŠTA-žema
pasuks visas detales tinkama montavimui kryptimi.

Paklauskime dabar, ar kiekvienas DBA yra sinchronizuojančiu automatu?

Atsakymas į šį klausimą yra neigiamas.

Tada paklauskime: kaip nustatyti, ar duotas DBA yra sinchronizuojantis ir kaip rasti šį automata sinchronizuojantį žodį w ?

Pasirodo, kad atsakyti į šį klausimą nėra labai sunku. Tik reikty duotam DBA sukonstruoti specialų baigtinį automata, vadinama **poaibių automatu**.

Parodysime, kaip tai padaryti.

Duotojo DBA $A = (Q, \Sigma, \delta)$ poaibių automato $P(A)$:

- būsenomis yra visi netušti aibės Q poaibiai,
- jei M yra kuris nors netuščias aibės Q poaibis ir $a \in \Sigma$, tai

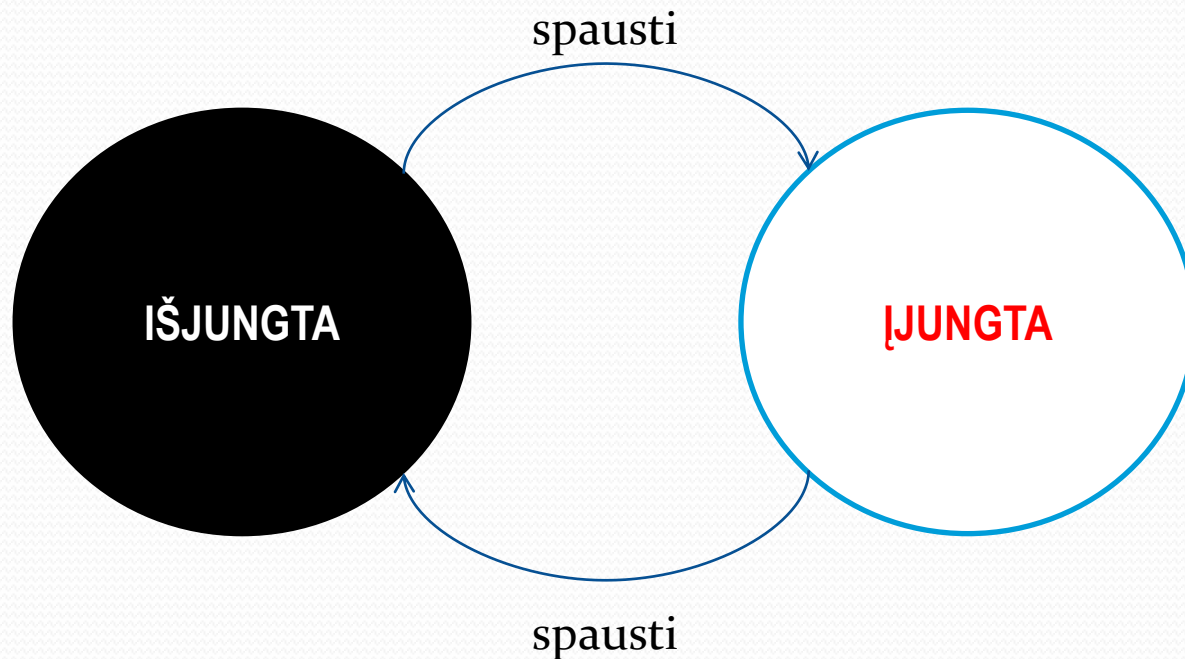
$$\delta(P, a) = \{\delta(m, a) : m \in M\}$$

Žodis $w \in \Sigma^*$ yra automata A sinchronizuojantis žodis tada ir tik tada, kai kelias poaibių automata $P(A)$ prasidedantis būsena Q ir pažymėtas žodžio w raidėmis baigiasi būsenoje pažymėtoje viena raide.

Paklauskime dabar, ar kiekvienas DBA yra sinchronizuojančiu automatu?

Atsakymas į šį klausimą yra **neigiamas**.

Pavyzdžiu galėtų būti paprasčiausias *jungiklio automatas*



Paklauskime dabar, ar kiekvienas DBA yra sinchronizuojančiu automatu?

Atsakymas į šį klausimą yra neigiamas.

Tada paklauskime: **kaip nustatyti, ar duotas DBA yra sinchronizuojantis ir kaip rasti šį automatą sinchronizuojantį žodį w ?**

Pasirodo, kad atsakyti į šį klausimą nėra labai sunku. Tik reikėtų duotam DBA sukonstruoti specialų baigtinį automatą, vadinamą **poaibių automatu**.

Parodysime, kaip tai padaryti.

Duotojo DBA $A = (Q, \Sigma, \delta)$ poaibių automato $P(A)$:

- būsenomis yra visi netušti aibės Q poaibiai,
- jei M yra kuris nors netuščias aibės Q poaibis ir $a \in \Sigma$, tai

$$\delta(P, a) = \{\delta(m, a) : m \in M\}$$

Žodis $w \in \Sigma^*$ yra automatą A sinchronizuojantis žodis tada ir tik tada, kai kelias poaibių automato $P(A)$ prasidedantis būsena Q ir pažymėtas žodžio w raidėmis baigiasi būsenoje pažymėtoje viena raide.

Paklauskime dabar, ar kiekvienas DBA yra sinchronizuojantis?

Atsakymas į šį klausimą yra neigiamas.

Tada paklauskime: kaip nustatyti, ar duotas DBA yra sinchronizuojantis ir kaip rasti šį automata sinchronizuojantį žodį w ?

Pasirodo, kad atsakyti į šį klausimą nėra labai sunku. Tik reiktų duotam DBA sukonstruoti specialų baigtinį automata, vadinamą **poaibių automatu**.

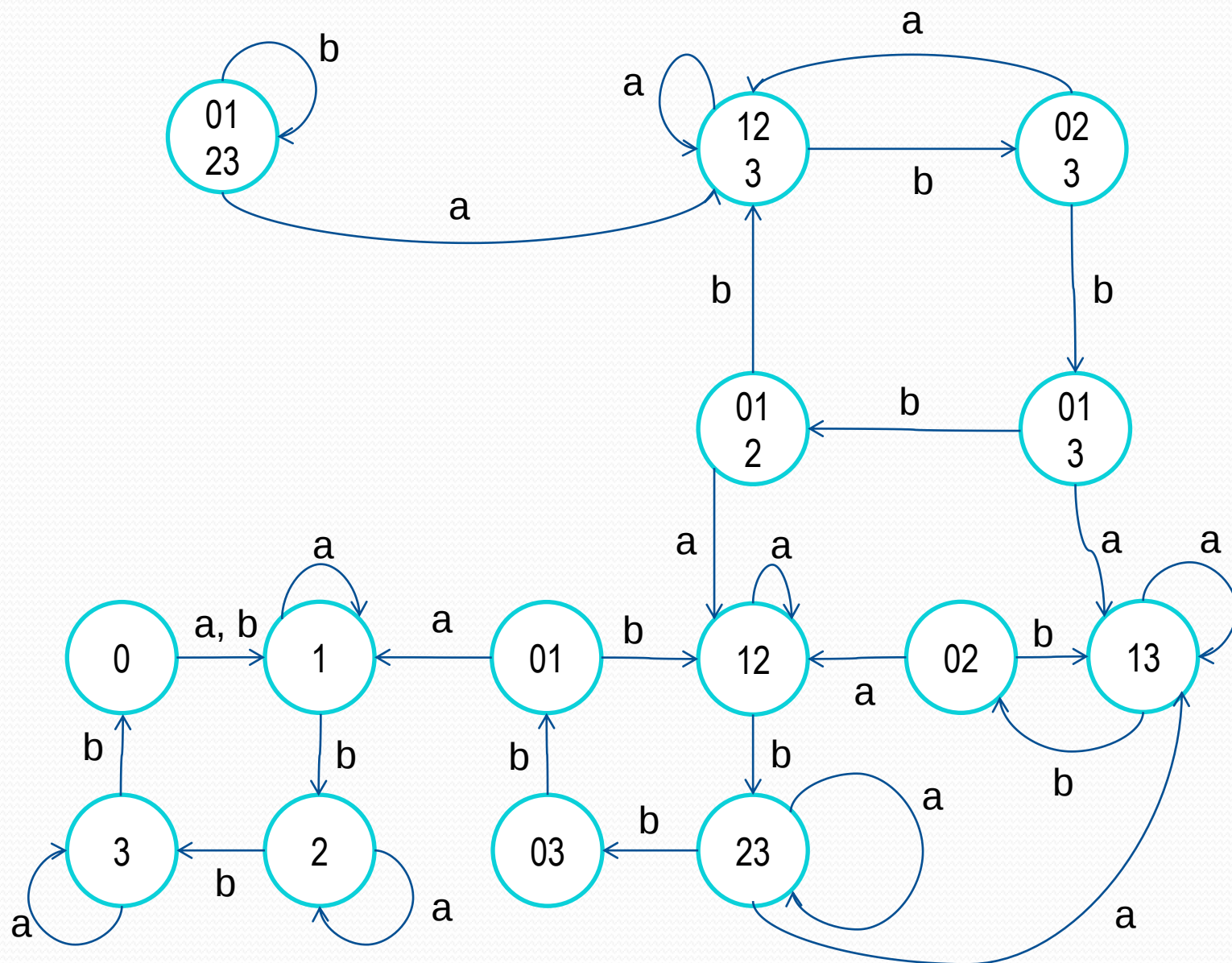
Parodysime, kaip tai padaryti.

Duotojo DBA $A = (Q, \Sigma, \delta)$ poaibių automato $P(A)$:

- būsenomis yra visi netušti aibės Q poaibiai,
- jei M yra kuris nors netuščias aibės Q poaibis ir $a \in \Sigma$, tai

$$\delta(P, a) = \{\delta(m, a) : m \in M\}$$

Žodis $w \in \Sigma^*$ yra automata A sinchronizuojantis žodis tada ir tik tada, kai kelias poaibių automate $P(A)$ prasidedantis būsena Q ir pažymėtas žodžio w raidėmis baigiasi būsenoje pažymėtoje viena raide.



Paklauskime dabar, ar kiekvienas DBA yra sinchronizuojantis?

Atsakymas į šį klausimą yra neigiamas.

Tada paklauskime: kaip nustatyti, ar duotas DBA yra sinchronizuojantis ir kaip rasti šį automata sinchronizuojantį žodį w ?

Pasirodo, kad atsakyti į šį klausimą nėra labai sunku. Tik reiktų duotam DBA sukonstruoti specialų baigtinį automata, vadinamą poaibių automatu.

Parodysime, kaip tai padaryti.

Duotojo DBA $A = (Q, \Sigma, \delta)$ **poaibių automato** $P(A)$:

- būsenomis yra visi *netušti* aibės Q poaibiai,
- jei $\{q_1, \dots, q_r\}$ yra kuris nors netuščias aibės Q poaibis ir $a \in \Sigma$, tai

$$\delta(\{q_1, \dots, q_r\}, a) = \{\delta(q_1, a), \dots, \delta(q_r, a)\}$$

Paklauskime dabar, ar kiekvienas DBA yra sinchronizuojantis?

Atsakymas į šį klausimą yra neigiamas.

Tada paklauskime: kaip nustatyti, ar duotas DBA yra sinchronizuojantis ir kaip rasti šį automata sinchronizuojantį žodį w ?

Pasirodo, kad atsakyti į šį klausimą nėra labai sunku. Tik reiktų duotam DBA sukonstruoti specialų baigtinį automata, vadinamą poaibių automatu.

Parodysime, kaip tai padaryti.

Duotojo DBA $A = (Q, \Sigma, \delta)$ poaibių automato $P(A)$:

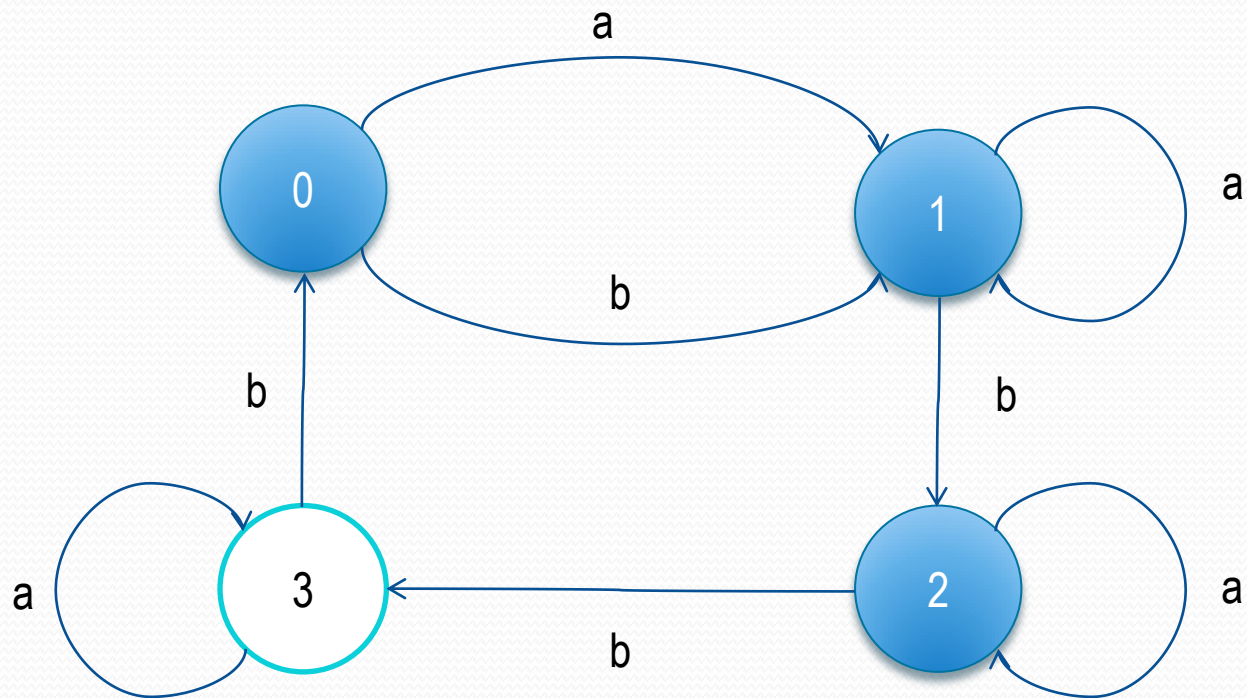
- būsenomis yra visi *netušti* aibės Q poaibiai,
- jei $\{q_1, \dots, q_r\}$ yra kuris nors netuščias aibės Q poaibis ir $a \in \Sigma$, tai

$$\delta(\{q_1, \dots, q_r\}, a) = \{\delta(q_1, a), \dots, \delta(q_r, a)\}$$

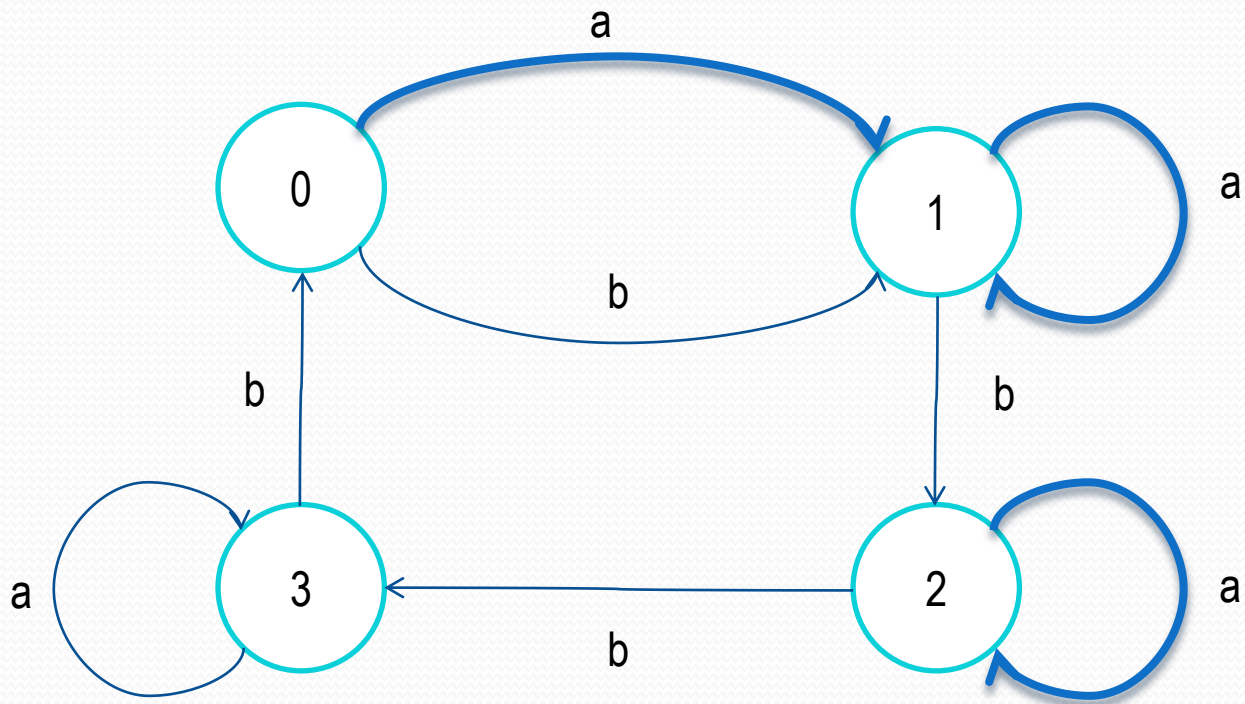
Jei automate A yra n būsenų, tai kiek būsenų yra poaibių automate $P(A)$?

Pavyzdžiui, mūsų automatui

$$\delta(\{0, 1, 2\}, a) = \{\delta(0, a), \delta(1, a), \delta(2, a)\} = \{1, 1, 2\} = \{1, 2\}$$

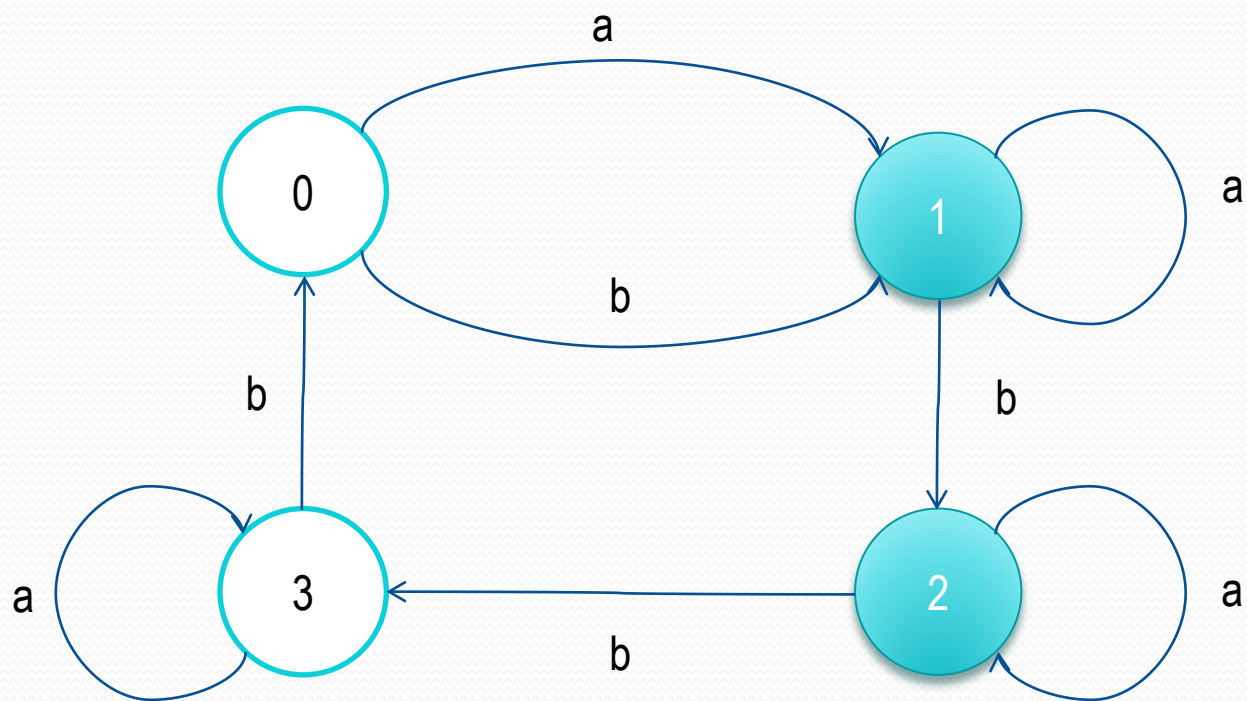


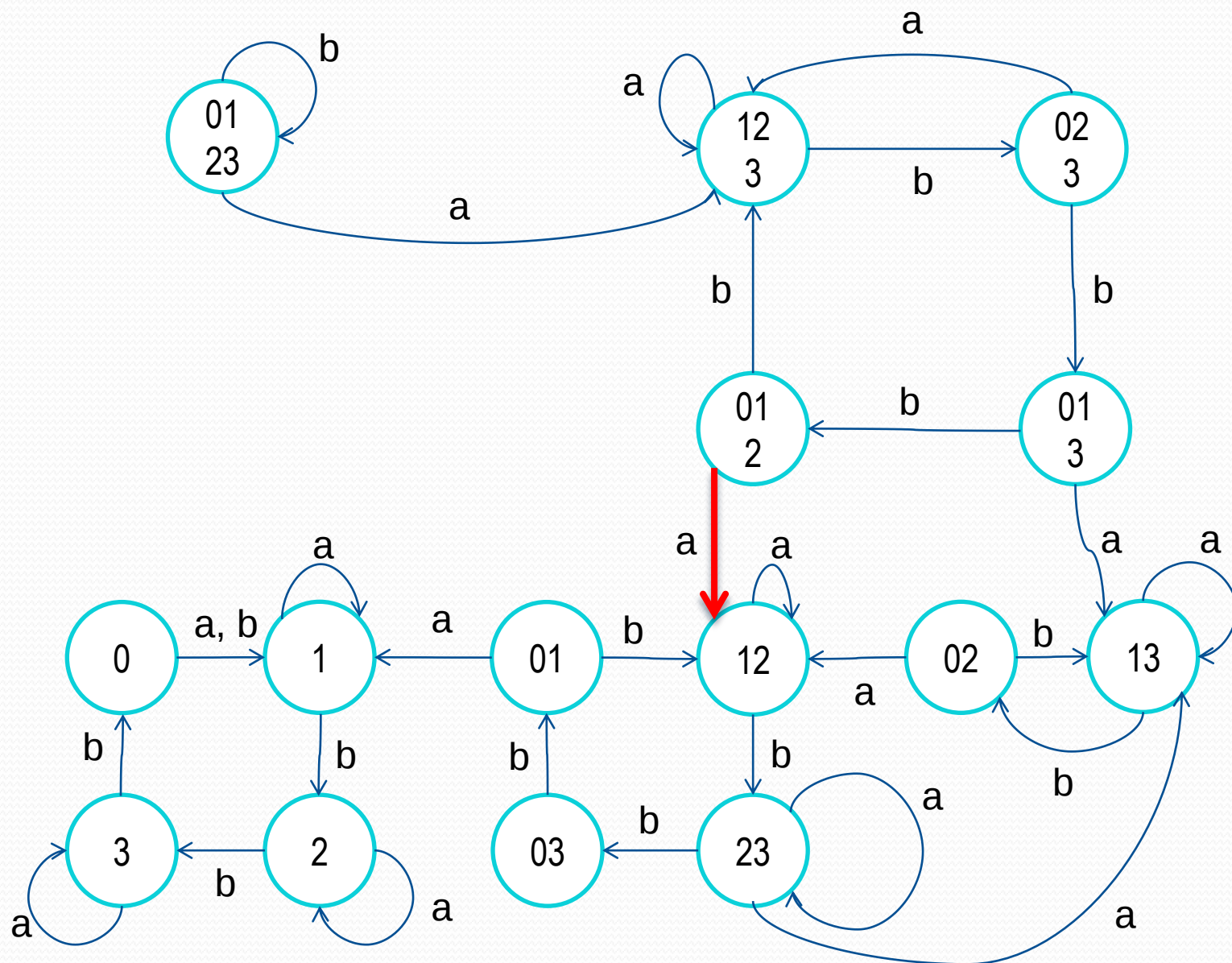
Pavyzdžiui, mūsų automatui

$$\delta(\{0, 1, 2\}, a) = \{\delta(0, a), \delta(1, a), \delta(2, a)\} = \{1, 1, 2\} = \{1, 2\}$$


Pavyzdžiui, mūsų automatui

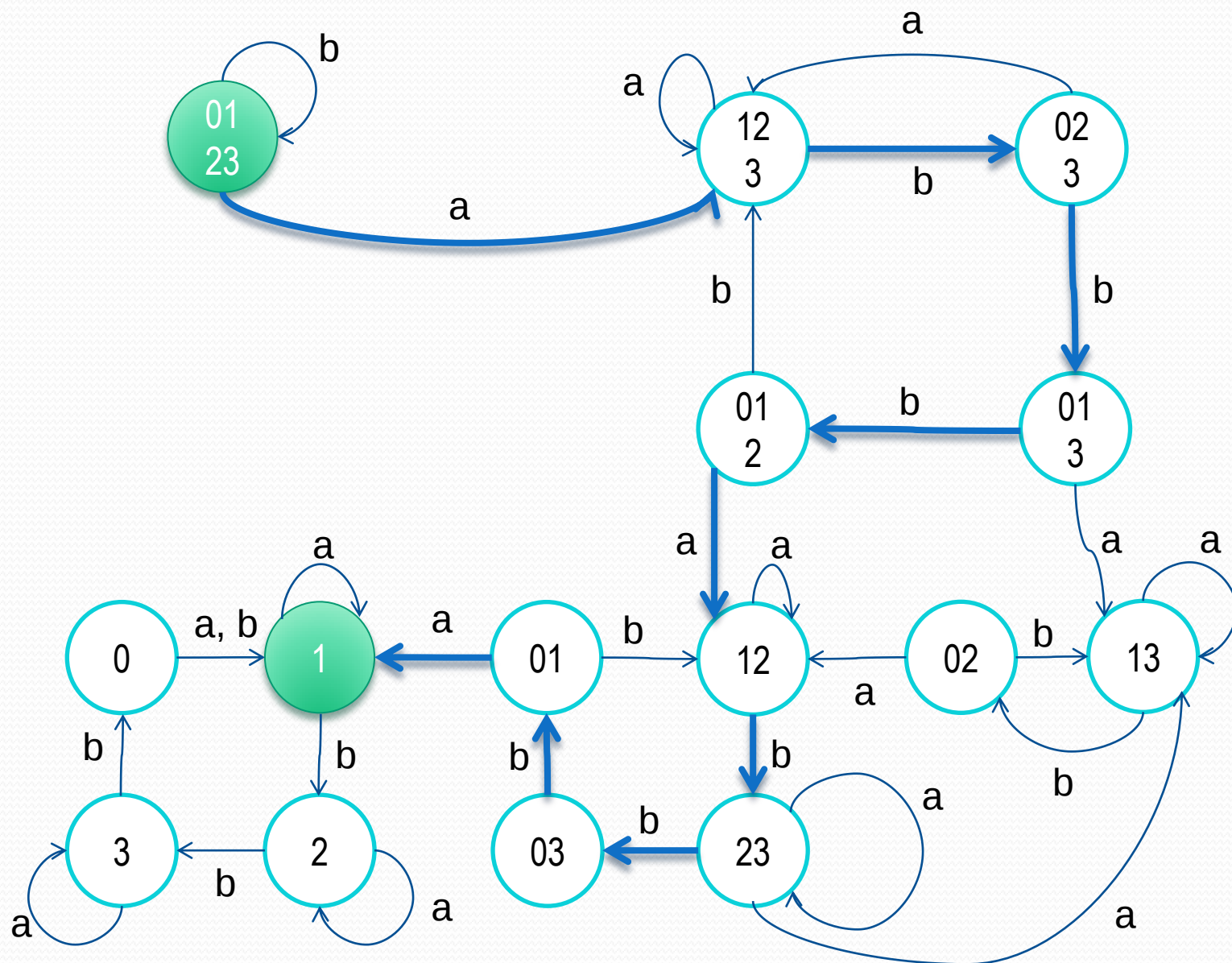
$$\delta(\{0, 1, 2\}, a) = \{\delta(0, a), \delta(1, a), \delta(2, a)\} = \{1, 1, 2\} = \{1, 2\}$$



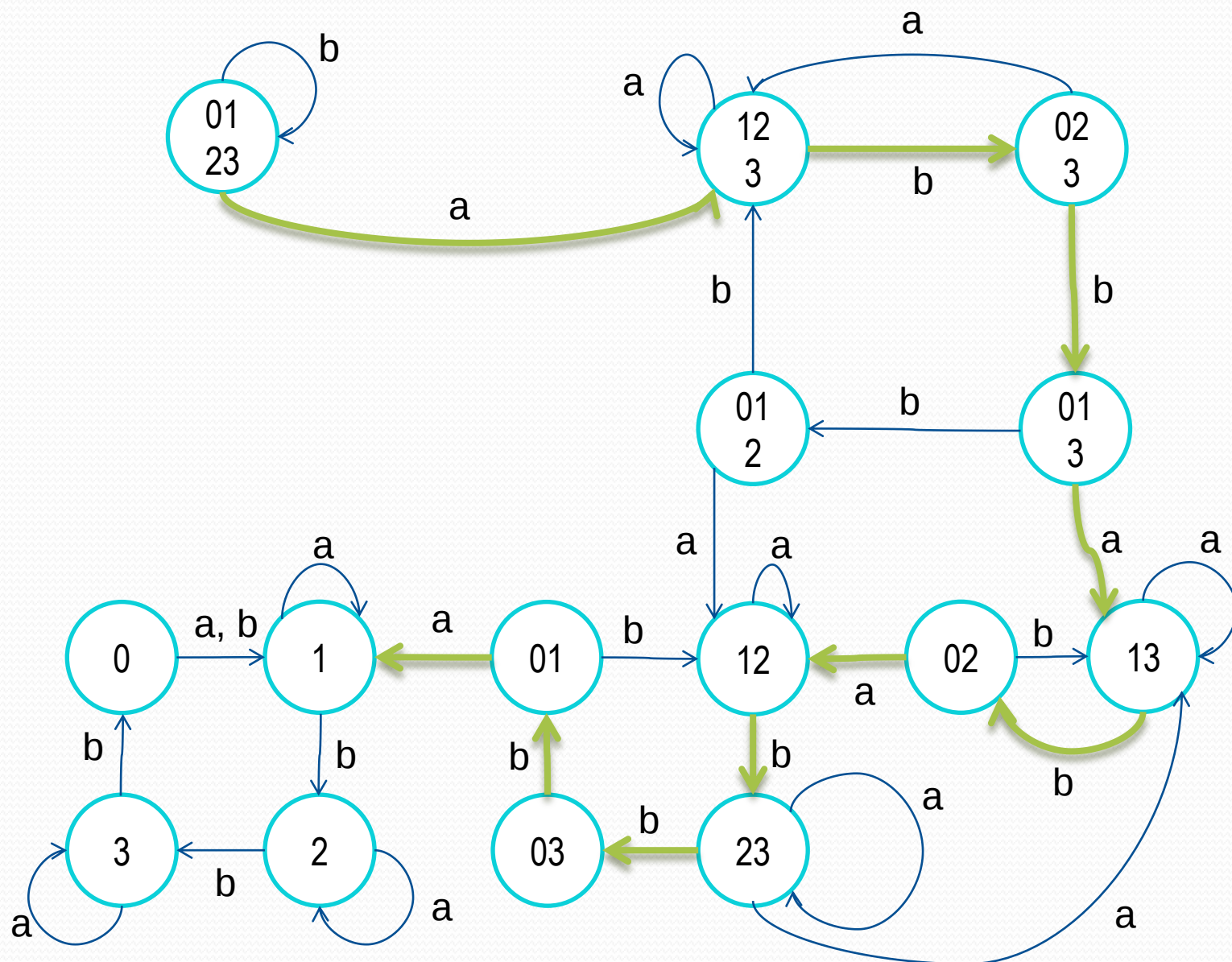


Sinchronizuojančio automato požymis

Žodis $w \in \Sigma^*$ yra automatą A sinchronizuojantis žodis tada ir tik tada, kai kelias poaibių automatai $P(A)$ prasidedantis būseną Q ir pažymėtas žodžio w raidėmis baigiasi būsenoje pažymėtoje viena raide.



Nesunku matyti, kad tai ne vienintelis DBA sinchronizuojantis žodis.
Kitas galėtų būti $w' = abbababbba$



Nesunku matyti, kad tai ne vienintelis DBA sinchronizuojantis žodis.

Kitas galėtų būti $w' = abbababbba$.

Todėl galėtume paklausti:

Koks yra mažiausias DBA sinchronizuojančio žodžio ilgis?

Atsakymas į šį klausimą daug sudėtingesnis.

Dar 1964 metais Jan Černý iškėlė hipotezę:

Kiekvienam sinchronizuojančiam DBA $A = (Q, \Sigma, \delta)$, turinčiam n būsenų, galima rasti $(n - 1)^2$ ilgio šį automata sinchronizuojantį žodį.

Nesunku matyti, kad tai ne vienintelis DBA sinchronizuojantis žodis.

Kitas galėtų būti $w' = abbababbba$.

Todėl galėtume paklausti:

Koks yra mažiausias DBA sinchronizuojančio žodžio ilgis?

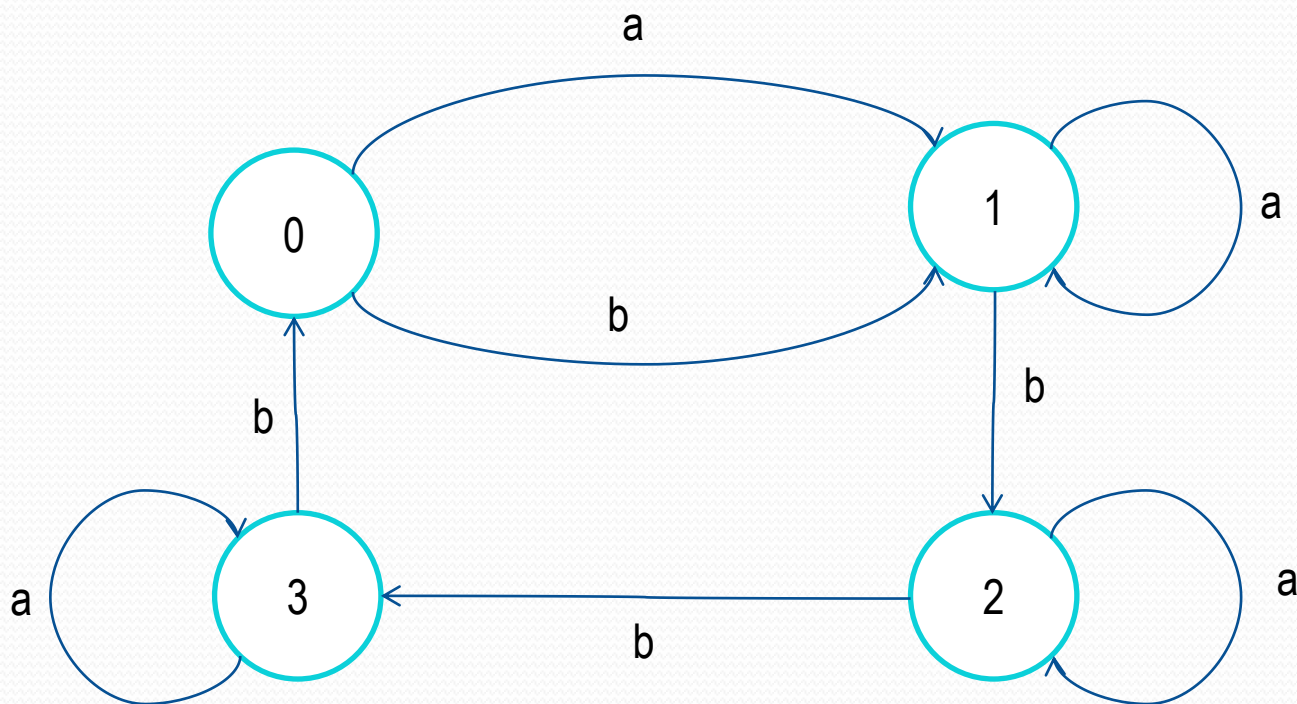
Atsakymas į šį klausimą daug sudėtingesnis.

Dar 1964 metais Jan Černý iškėlė **hipotezę**:

Kiekvienam sinchronizuojančiam DBA $A = (Q, \Sigma, \delta)$, turinčiam n būsenų, galima rasti $(n - 1)^2$ ilgio šį automatą sinchronizuojantį žodį.

Tais pačiais metais mokslininkas sukonstravo sinchronizuojančių automatų, tenkinančių hipotezę, seką.

Vienas iš tokių automatų ir yra mūsų jau nagrinėtas 4 būsenų automatas, turintis $(4-1)^2 = 9$ raidžių jį sinchronizuojantį žodį $w = abbbabbba$



Nuo to laiko daug kartų buvo bandyta arba įrodyti, arba paneigti taip lengvai formuluojamą hipotezę, bet iki šiol niekam tai nepavyko.

Šiuo metu hipotezė yra patvirtinta dideliame automatų ratui, bet bendru atveju yra gautas tik toks **viršutinis** sinchronizuojančio žodžio ilgio įvertis: $\frac{n^3 - n}{3}$.

Taigi, n būsenų turinčio DBA sinchronizuojančio žodžio ilgis i yra intervale

$$(n - 1)^2 \leq i \leq \frac{n^3 - n}{3}.$$

Černų hipotezė dar ir šiandien laukia savo įrodymo!

Nuo to laiko daug kartų buvo bandyta arba įrodyti, arba paneigti taip lengvai formuluojamą hipotezę, bet iki šiol niekam tai nepavyko.

Šiuo metu hipotezė yra patvirtinta dideliame automatų ratui, bet bendru atveju yra gautas tik toks viršutinis sinchronizuojančio žodžio ilgio įvertis: $\frac{n^3 - n}{3}$.

Taigi, n būsenų turinčio DBA sinchronizuojančio žodžio ilgis i yra intervale

$$(n - 1)^2 \leq i \leq \frac{n^3 - n}{3}.$$

Černý hipotezė dar ir šiandien laukia savo įrodymo!