

# Nowe życie starych automatów

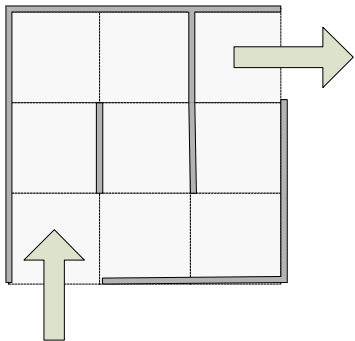
Paweł Rzążewski

MiNI Akademia Matematyki

14.04.2012

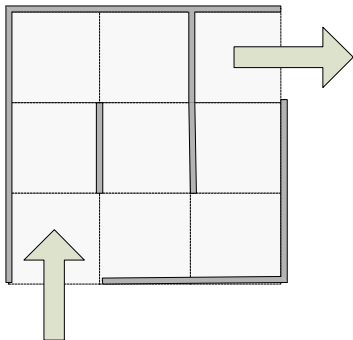
# Przykład 1 – labirynt

Chcemy przeprowadzić robota przez labirynt. Dysponujemy kontrolerem z czterema przyciskami ( $\uparrow$ ,  $\downarrow$ ,  $\leftarrow$ ,  $\rightarrow$ ), które pozwalają poruszać robotem w jednym z czterech kierunków.



## Przykład 1 – labirynt

Chcemy przeprowadzić robota przez labirynt. Dysponujemy kontrolerem z czterema przyciskami ( $\uparrow$ ,  $\downarrow$ ,  $\leftarrow$ ,  $\rightarrow$ ), które pozwalają poruszać robotem w jednym z czterech kierunków.

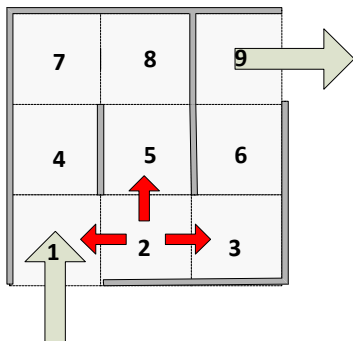


Poprawna jest np. sekwencja  $\rightarrow\rightarrow\uparrow\uparrow$ , ale też  $\uparrow\uparrow\rightarrow\downarrow\downarrow\rightarrow\leftarrow\rightarrow\uparrow\uparrow$  (niektóre pola odwiedzamy więcej niż raz).

Które sekwencje pozwalają na dojście do wyjścia?

# Przykład 1 – labirynt

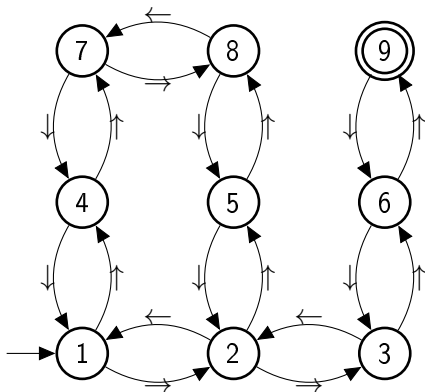
Z każdym polem możemy skojarzyć możliwe ruchy i pola, do których one prowadzą.



## Przykład 1 – labirynt

Z każdym polem możemy skojarzyć możliwe ruchy i pola, do których one prowadzą.

Wtedy labirynt możemy zapisać jako zbiór możliwych stanów i przejść między stanami.



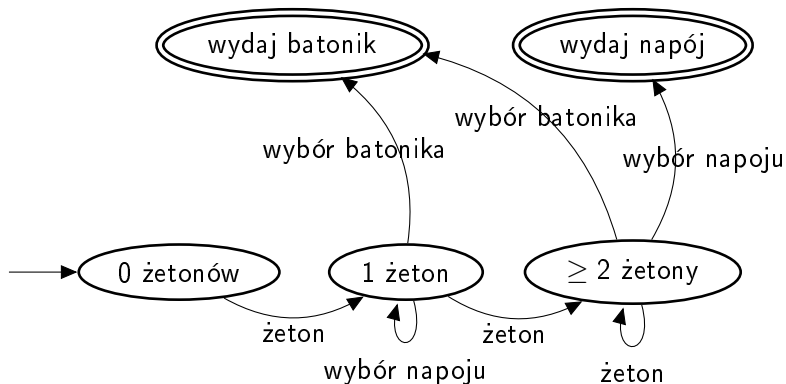
Stan 1 jest stanem *wejściowym*, a stan 9 jest stanem *akceptującym*.

## Przykład 2 – automat ze słodyczami

Automat wymienia żetony na słodycze. Batonik kosztuje 1 żeton, a napój 2 żetony. Automat nie wydaje reszty.

## Przykład 2 – automat ze słodyczami

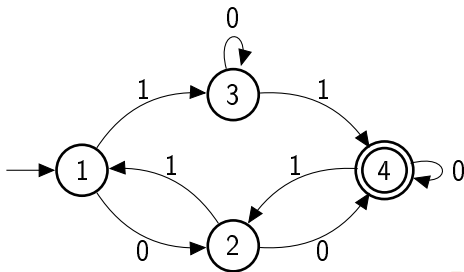
Automat wymienia żetony na słodycze. Batonik kosztuje 1 żeton, a napój 2 żetony. Automat nie wydaje reszty.



# Deterministyczny automat skończony

Automat jest abstrakcyjną maszyną, którą można sterować wysyłając symbole sterujące z pewnego skończonego zbioru, zwanego *alfabetem*. Automat składa się z:

- 1 stanów, w tym
  - dokładnie jednego stanu wejściowego (ozn. strzałką)
  - co najmniej jednego stanu akceptującego (ozn. podwójną linią)
- 2 przejść między stanami (przejście jest w postaci: jeśli znajdujesz się w stanie  $x$  i wczytasz symbol  $a$ , przejdź do stanu  $y$ ).





- Automaty skończone były od dawna używane (w sposób nieformalny) przy konstrukcji urządzeń elektrycznych i mechanicznych.

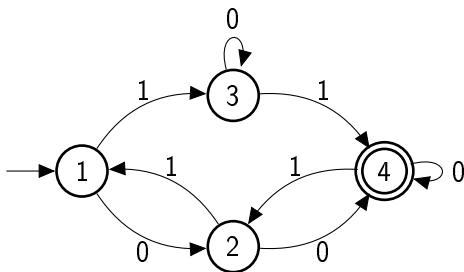
- Automaty skończone były od dawna używane (w sposób nieformalny) przy konstrukcji urządzeń elektrycznych i mechanicznych.
- Pierwszy formalny opis automatów pochodzi z roku 1943. Zawdzięczamy go dwóm neurobiologom(!) McCullochowi i Pittsowi. Automaty służyły do modelowania komórek w mózgu człowieka.

- Automaty skończone były od dawna używane (w sposób nieformalny) przy konstrukcji urządzeń elektrycznych i mechanicznych.
- Pierwszy formalny opis automatów pochodzi z roku 1943. Zawdzięczamy go dwóm neurobiologom(!) McCullochowi i Pittsowi. Automaty służyły do modelowania komórek w mózgu człowieka.
- Prawdziwą karierę automaty zaczęły wraz z początkiem ery komputerów.

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

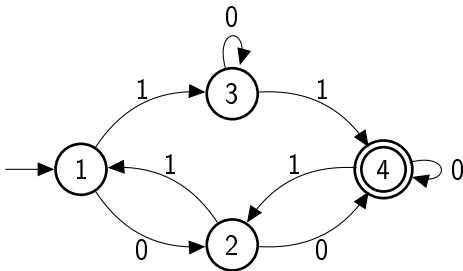
- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie



# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

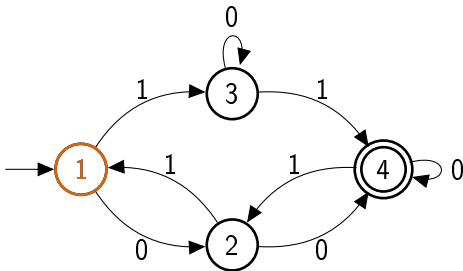


Czy automat akceptuje słowo 101010?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

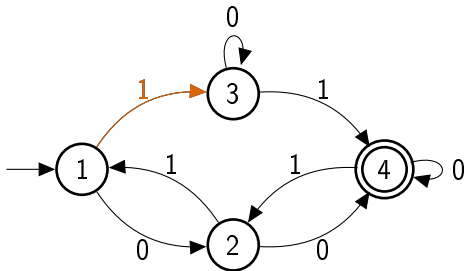


Czy automat akceptuje słowo 101010?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

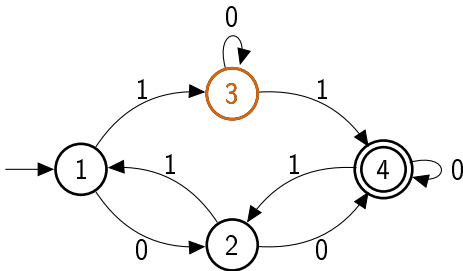


Czy automat akceptuje słowo 101010?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie



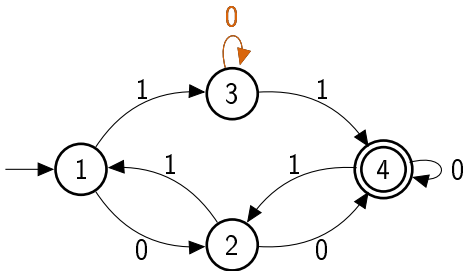
Czy automat akceptuje słowo 101010?



# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

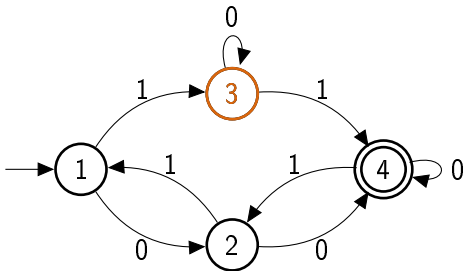


Czy automat akceptuje słowo 101010?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

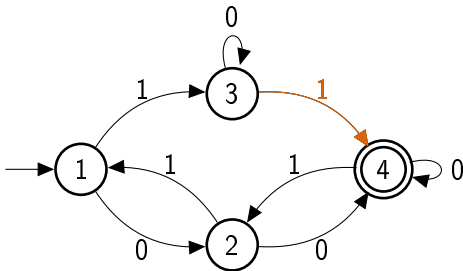


Czy automat akceptuje słowo 101010?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

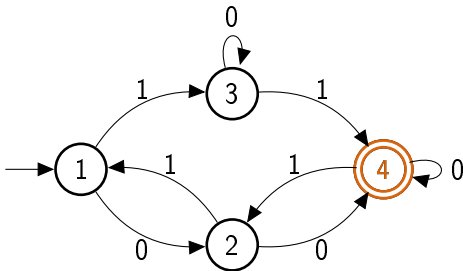


Czy automat akceptuje słowo 101010?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

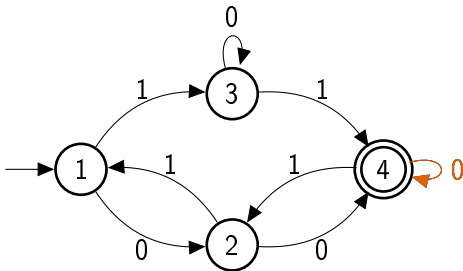


Czy automat akceptuje słowo 101010?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

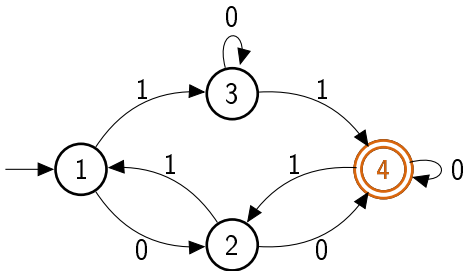


Czy automat akceptuje słowo 101010?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

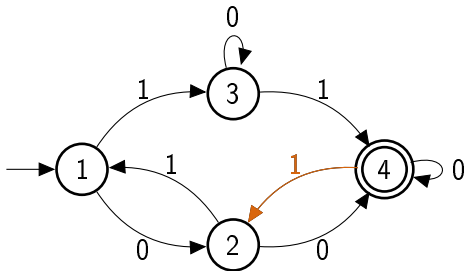


Czy automat akceptuje słowo 101010?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

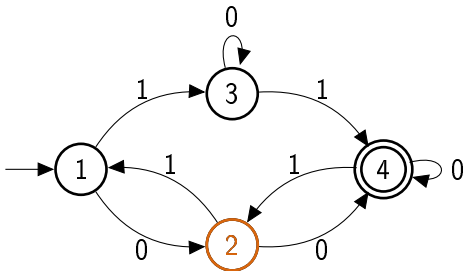


Czy automat akceptuje słowo 101010?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie



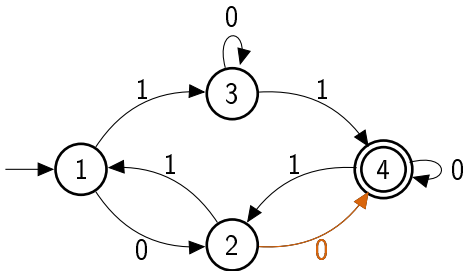
Czy automat akceptuje słowo 101010?



# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

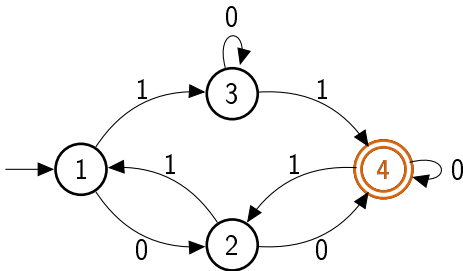


Czy automat akceptuje słowo 101010?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

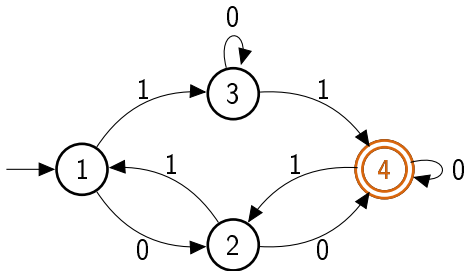


Czy automat akceptuje słowo 101010?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

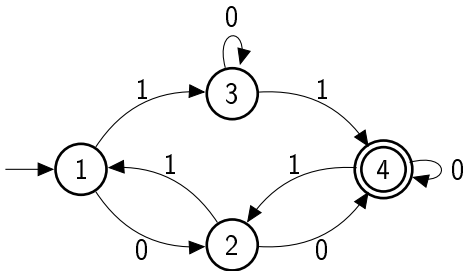


Czy automat akceptuje słowo 101010? **TAK**

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

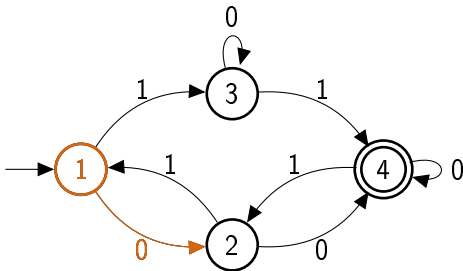


Czy automat akceptuje słowo 0101?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

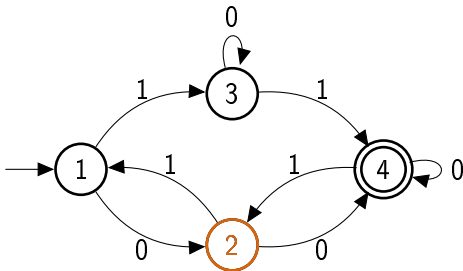


Czy automat akceptuje słowo 0101?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

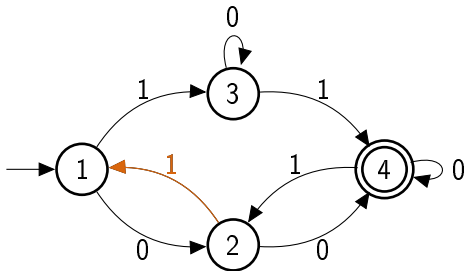


Czy automat akceptuje słowo 0101?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

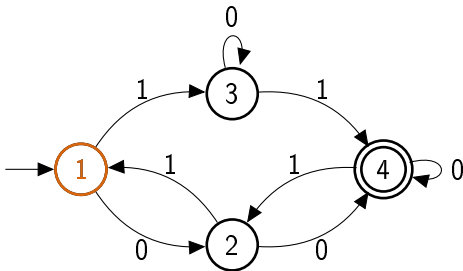


Czy automat akceptuje słowo 0101?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie



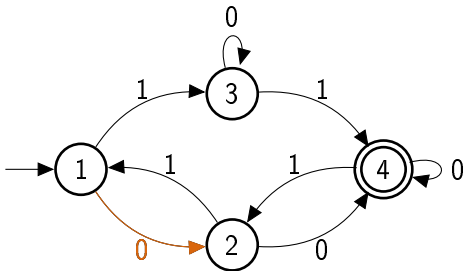
Czy automat akceptuje słowo 0101?



# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

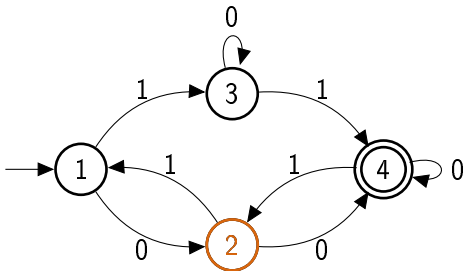


Czy automat akceptuje słowo 0101?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

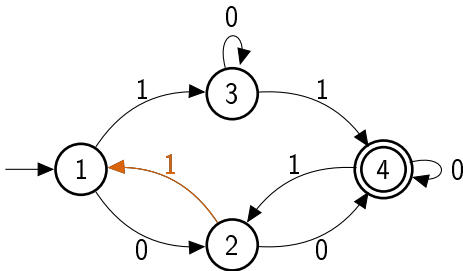


Czy automat akceptuje słowo 0101?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

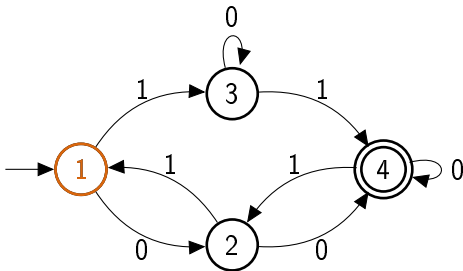


Czy automat akceptuje słowo 0101?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie

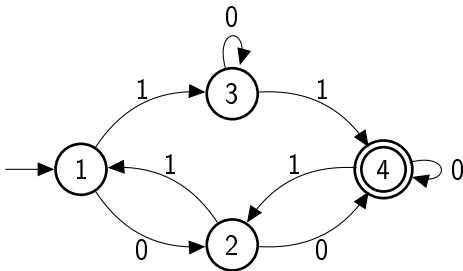


Czy automat akceptuje słowo 0101?

# Słowa akceptowane przez automat

Jeśli  $\Sigma$  jest alfabetem danego automatu, a  $w$  słowem (czyli ciągiem znaków) z alfabetu  $\Sigma$ , mówimy, że:

- automat *akceptuje* słowo  $w$ , jeśli po jego wczytaniu znajduje się w jednym ze stanów akceptujących
- automat *odrzuca* słowo  $w$ , jeśli po jego wczytaniu znajduje się w innym stanie



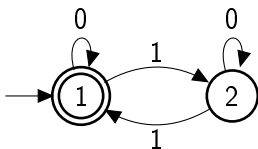
Czy automat akceptuje słowo 0101? **NIE**

Zbiór słów akceptowanych przez automat nazywamy *językiem* akceptowanym przez automat. Język ten może być skończony lub nieskończony.

# Język akceptowany przez automat

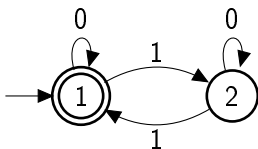
Zbiór słów akceptowanych przez automat nazywamy *językiem* akceptowanym przez automat. Język ten może być skończony lub nieskończony.

Jaki jest język akceptowany przez poniższy automat?



Zbiór słów akceptowanych przez automat nazywamy *językiem* akceptowanym przez automat. Język ten może być skończony lub nieskończony.

Jaki jest język akceptowany przez poniższy automat?



Automat ten akceptuje wszystkie słowa nad alfabetem  $\{0, 1\}$ , w których występuje parzysta liczba jedynek.



Wyrażenia regularne są wygodnym sposobem opisu języków akceptowanych przez automaty.

Wyrażenia regularne składają się z symboli języka i trzech operatorów:

Wyrażenia regularne są wygodnym sposobem opisu języków akceptowanych przez automaty.

Wyrażenia regularne składają się z symboli języka i trzech operatorów:

- 1 Sumy (|) – oznaczającej alternatywę

Wyrażenia regularne są wygodnym sposobem opisu języków akceptowanych przez automaty.

Wyrażenia regularne składają się z symboli języka i trzech operatorów:

- 1 Sumy ( $|$ ) – oznaczającej alternatywę
- 2 Kontatenacji (czyli sklejenia)

Wyrażenia regularne są wygodnym sposobem opisu języków akceptowanych przez automaty.

Wyrażenia regularne składają się z symboli języka i trzech operatorów:

- 1 Sumy ( $|$ ) – oznaczającej alternatywę
- 2 Kontatenacji (czyli sklejenia)
  - $aa(a|b|c)$  oznacza, że w języku znajdują się słowa  $aaa$ ,  $aab$ ,  $aac$

Wyrażenia regularne są wygodnym sposobem opisu języków akceptowanych przez automaty.

Wyrażenia regularne składają się z symboli języka i trzech operatorów:

- 1 Sumy ( $|$ ) – oznaczającej alternatywę
- 2 Kontatenacji (czyli sklejenia)
  - $aa(a|b|c)$  oznacza, że w języku znajdują się słowa  $aaa$ ,  $aab$ ,  $aac$
- 3 Gwiazdki Kleena ( $*$ ) – oznaczającej powtórzenie danego ciągu dowolną liczbę razy (w tym 0!)

Wyrażenia regularne są wygodnym sposobem opisu języków akceptowanych przez automaty.

Wyrażenia regularne składają się z symboli języka i trzech operatorów:

- 1 Sumy ( $|$ ) – oznaczającej alternatywę
- 2 Kontatenacji (czyli sklejenia)
  - $aa(a|b|c)$  oznacza, że w języku znajdują się słowa  $aaa$ ,  $aab$ ,  $aac$
- 3 Gwiazdki Kleena ( $*$ ) – oznaczającej powtórzenie danego ciągu dowolną liczbę razy (w tym 0!)
  - $(ab)^*$  oznacza, że w języku znajdują się słowa: puste (ozn. przez  $\epsilon$ ),  $ab$ ,  $abab$ ,  $ababab$  itd.

Wyrażenia regularne są wygodnym sposobem opisu języków akceptowanych przez automaty.

Wyrażenia regularne składają się z symboli języka i trzech operatorów:

- 1 Sumy ( $|$ ) – oznaczającej alternatywę
- 2 Kontatenacji (czyli sklejenia)
  - $aa(a|b|c)$  oznacza, że w języku znajdują się słowa  $aaa$ ,  $aab$ ,  $aac$
- 3 Gwiazdki Kleena ( $*$ ) – oznaczającej powtórzenie danego ciągu dowolną liczbę razy (w tym 0!)
  - $(ab)^*$  oznacza, że w języku znajdują się słowa: puste (ozn. przez  $\epsilon$ ),  $ab$ ,  $abab$ ,  $ababab$  itd.

Więcej o konstruowaniu wyrażeń regularnych będzie na warsztatach.

Jaki język jest opisywany przez poniższe wyrażenie regularne?

$$(0|1|\dots|9)(0|1|\dots|9) - (0|1|\dots|9)(0|1|\dots|9)(0|1|\dots|9)$$



Jaki język jest opisywany przez poniższe wyrażenie regularne?

$$(0|1|\dots|9)(0|1|\dots|9) - (0|1|\dots|9)(0|1|\dots|9)(0|1|\dots|9)$$

Są to napisy w postaci XX-XXX, gdzie pod znakiem X może być dowolna cyfra.

Jaki język jest opisywany przez poniższe wyrażenie regularne?

$$(0|1|\dots|9)(0|1|\dots|9) - (0|1|\dots|9)(0|1|\dots|9)(0|1|\dots|9)$$

Są to napisy w postaci XX-XXX, gdzie pod znakiem X może być dowolna cyfra. Kody pocztowe w Polsce mają taką właśnie postać.

Jaki język jest opisywany przez poniższe wyrażenie regularne?

$$(0|1|\dots|9)(0|1|\dots|9) - (0|1|\dots|9)(0|1|\dots|9)(0|1|\dots|9)$$

Są to napisy w postaci XX-XXX, gdzie pod znakiem X może być dowolna cyfra. Kody pocztowe w Polsce mają taką właśnie postać.

Wyrażenia regularne (i automaty) stosowane są powszechnie przy sprawdzaniu poprawności wpisanych danych (np. w formularzach na stronie internetowej, kompilatorach itp.).

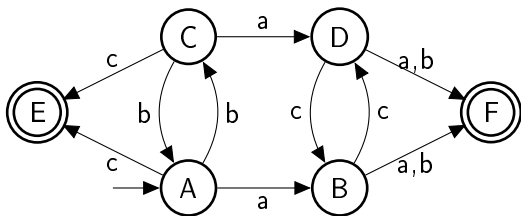
Jaki język jest opisywany przez poniższe wyrażenie regularne?

$$(0|1|\dots|9)(0|1|\dots|9) - (0|1|\dots|9)(0|1|\dots|9)(0|1|\dots|9)$$

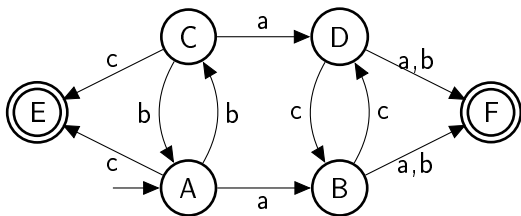
Są to napisy w postaci XX-XXX, gdzie pod znakiem X może być dowolna cyfra. Kody pocztowe w Polsce mają taką właśnie postać.

Wyrażenia regularne (i automaty) stosowane są powszechnie przy sprawdzaniu poprawności wpisanych danych (np. w formularzach na stronie internetowej, kompilatorach itp.).

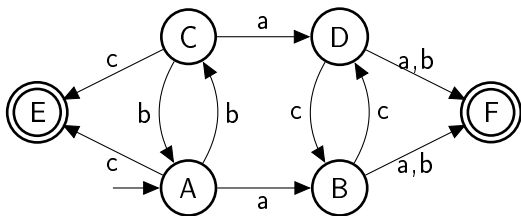
Jakie wyrażenie regularne opisuje adres emailowy?



Jaki język akceptuje ten automat?



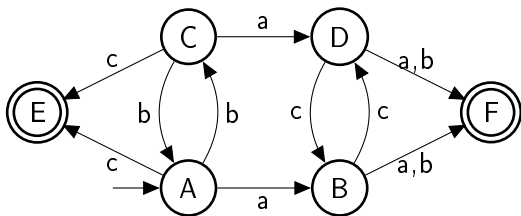
Jaki język akceptuje ten automat?  $(bb)^*ac^*|(bb)^*bac^*$



Jaki język akceptuje ten automat?  $(bb)^*ac^*|(bb)^*bac^*$

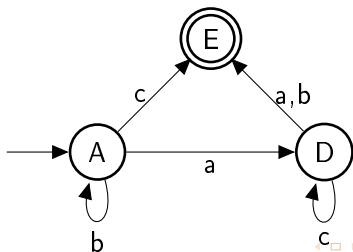
Ale kiedy przyjrzymy się bliżej, okaże się, że można to wyrażenie uprościć do postaci  $b^*ac^*$ .

# Wyrażenia regularne



Jaki język akceptuje ten automat?  $(bb)^*ac^*|(bb)^*bac^*$

Ale kiedy przyjrzymy się bliżej, okaże się, że można to wyrażenie uprościć do postaci  $b^*ac^*$ .





Języki, które mogą być opisane wyrażeniem regularnym nazywamy *regularnymi*. Czy istnieją języki nieregularne?

Języki, które mogą być opisane wyrażeniem regularnym nazywamy *regularnymi*. Czy istnieją języki nieregularne?

Oczywiście :).

Języki, które mogą być opisane wyrażeniem regularnym nazywamy *regularnymi*. Czy istnieją języki nieregularne?

Oczywiście :).

Rozważmy język składający się ze słów w postaci

$\underbrace{000\dots0}_k \underbrace{111\dots1}_k$  ( $k \geq 0$ ).

Nietrudno zauważyć, że nie da się skonstruować automatu skończonego, który akceptowałby taki język.

Języki, które mogą być opisane wyrażeniem regularnym nazywamy *regularnymi*. Czy istnieją języki nieregularne?

Oczywiście :).

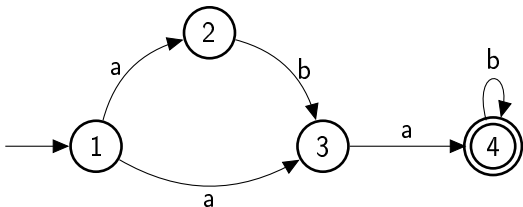
Rozważmy język składający się ze słów w postaci

$\underbrace{000 \dots 0}_k \underbrace{111 \dots 1}_k$  ( $k \geq 0$ ).

Nietrudno zauważyć, że nie da się skonstruować automatu skończonego, który akceptowałby taki język.

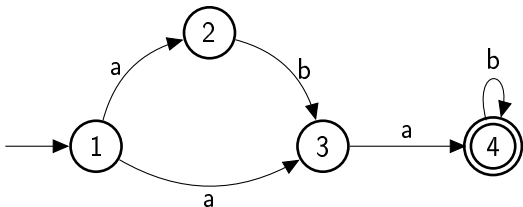
# Niedeterministyczny automat skończony

Przypuśćmy, że nasz automat po wczytaniu symbolu ma możliwość wyboru jednego z kilku przejść...



# Niedeterministyczny automat skończony

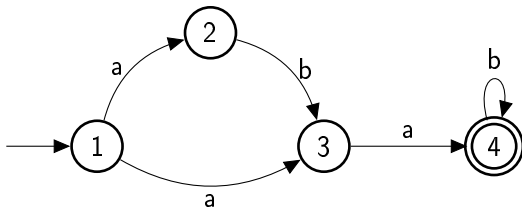
Przypuśćmy, że nasz automat po wczytaniu symbolu ma możliwość wyboru jednego z kilku przejść...



Automat akceptuje dane słowo, jeżeli istnieje taki wybór dozwolonych przejść, że automat zakończy wczytywanie słowa w stanie akceptującym.

# Niedeterministyczny automat skończony

Przypuśćmy, że nasz automat po wczytaniu symbolu ma możliwość wyboru jednego z kilku przejść...



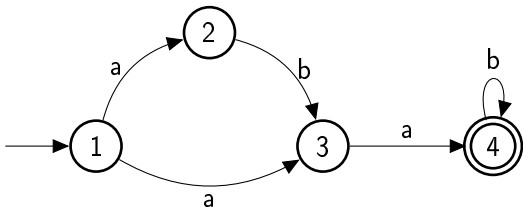
Automat akceptuje dane słowo, jeżeli istnieje taki wybór dozwolonych przejść, że automat zakończy wczytywanie słowa w stanie akceptującym.

Ponadto, jeżeli np. w stanie 1 automat wczyta symbol  $b$ , od razu odrzuca całe słowo.

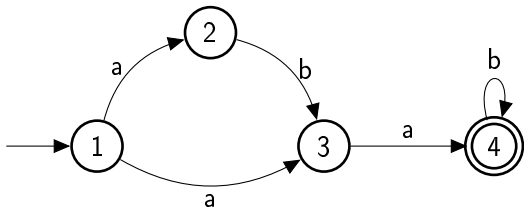
Jakie języki są akceptowane przez automaty niedeterministyczne?



Jakie języki są akceptowane przez automaty niedeterministyczne?

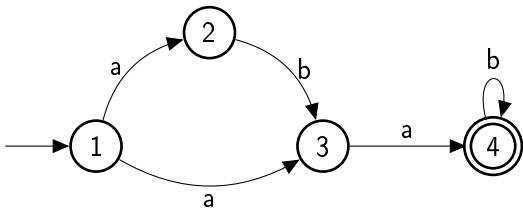


Jakie języki są akceptowane przez automaty niedeterministyczne?



Język dla tego automatu opisuje wyrażenie  $(ab|a)ab^*$ .

Jakie języki są akceptowane przez automaty niedeterministyczne?



Język dla tego automatu opisuje wyrażenie  $(ab|a)ab^*$ .

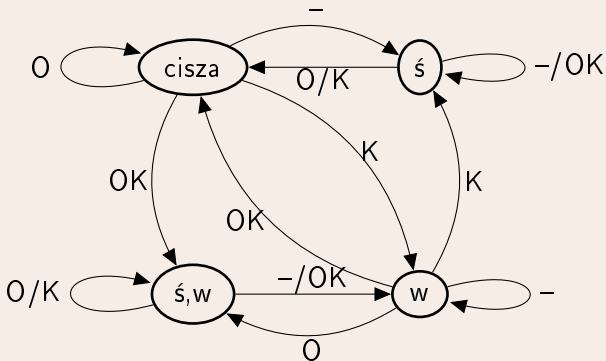
Języki akceptowane przez automaty niedeterministyczne to języki regularne – czyli dla każdego automatu niedeterministycznego istnieje automat deterministyczny, akceptujący ten sam język!

Ross Ashby, 1956

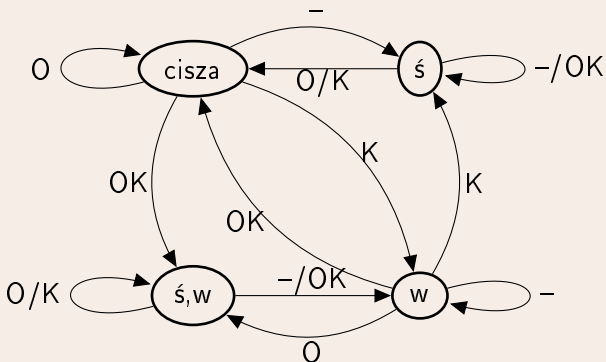
Pewien bardzo stary dom jest nawiedzany przez duchy – słychać w nim śmiech i grobowe wycie. Zauważono, że granie na organach i zapalenie kadzidła wpływa na zachowanie duchów.

Ross Ashby, 1956

Pewien bardzo stary dom jest nawiedzany przez duchy – słysząc w nim Śmiech i grobowe Wycie. Zauważono, że granie na Organach i zapalanie Kadzidła wpływa na zachowanie duchów. (brak akcji oznaczamy przez -)

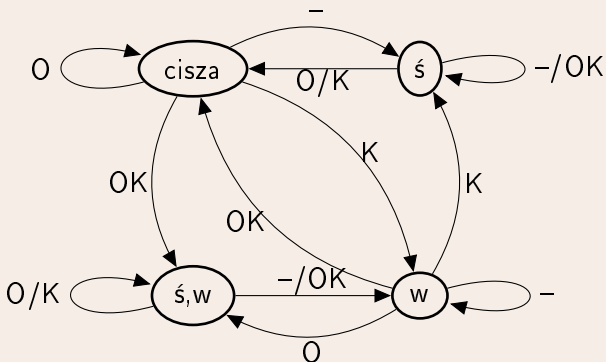


Ross Ashby, 1956



Znajdź sekwencję akcji, które uciszą duchy, niezależnie co aktualnie robią.

Ross Ashby, 1956



Znajdź sekwencję akcji, które uciszą duchy, niezależnie co aktualnie robią.

Taką sekwencją jest na przykład **- ,OK,0**.

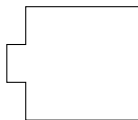
*Słowem synchronizującym* dla danego automatu nazywamy takie słowo, że niezależnie z którego stanu rozpoczniemy wczytywanie słowa, skończymy zawsze w tym samym stanie.

Automat, który ma słowo synchronizujące, nazywamy *synchronizowalnym*.



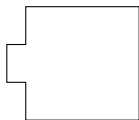
# Automaty synchronizowalne w robotyce

Jeden z elementów używanych na linii produkcyjnej ma następujący kształt:

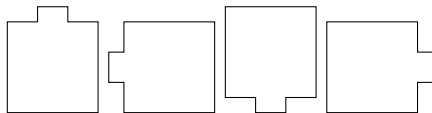


# Automaty synchronizowalne w robotyce

Jeden z elementów używanych na linii produkcyjnej ma następujący kształt:

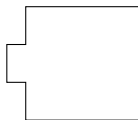


Elementy powinny zostać posortowane i poprawnie ustawione przed wysłaniem ich na linię produkcyjną. Początkowo element może znajdować się w jednej z pozycji:

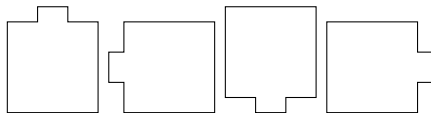


# Automaty synchronizowalne w robotyce

Jeden z elementów używanych na linii produkcyjnej ma następujący kształt:



Elementy powinny zostać posortowane i poprawnie ustawione przed wysłaniem ich na linię produkcyjną. Początkowo element może znajdować się w jednej z pozycji:



W jaki sposób automatycznie ustawić te elementy we właściwej pozycji?

# Automaty synchronizowalne w robotyce

Dobre rozwiązanie powinno być:

# Automaty synchronizowalne w robotyce

Dobre rozwiązanie powinno być:

- zawsze skuteczne

# Automaty synchronizowalne w robotyce

Dobre rozwiązanie powinno być:

- zawsze skuteczne
- proste (i co za tym idzie, tanie)

Dobre rozwiązanie powinno być:

- zawsze skuteczne
- proste (i co za tym idzie, tanie)
- szybkie – chcemy uniknąć długiego zajmowania się jednym elementem

# Automaty synchronizowalne w robotyce

Dobre rozwiązanie powinno być:

- zawsze skuteczne
- proste (i co za tym idzie, tanie)
- szybkie – chcemy uniknąć długiego zajmowania się jednym elementem

Elementy będą nadchodzić po kolei na taśmie produkcyjnej.  
Będziemy używali dwóch typów przeszkód:



Dobre rozwiązanie powinno być:

- zawsze skuteczne
- proste (i co za tym idzie, tanie)
- szybkie – chcemy uniknąć długiego zajmowania się jednym elementem

Elementy będą nadchodzić po kolei na taśmie produkcyjnej.

Będziemy używali dwóch typów przeszkód:

- Wysokiej

Dobre rozwiązanie powinno być:

- zawsze skuteczne
- proste (i co za tym idzie, tanie)
- szybkie – chcemy uniknąć długiego zajmowania się jednym elementem

Elementy będą nadchodzić po kolei na taśmie produkcyjnej.  
Będziemy używali dwóch typów przeszkód:

- Wysokiej



Dobre rozwiązanie powinno być:

- zawsze skuteczne
- proste (i co za tym idzie, tanie)
- szybkie – chcemy uniknąć długiego zajmowania się jednym elementem

Elementy będą nadchodzić po kolei na taśmie produkcyjnej.  
Będziemy używali dwóch typów przeszkód:

- Wysokiej



Dobre rozwiązanie powinno być:

- zawsze skuteczne
- proste (i co za tym idzie, tanie)
- szybkie – chcemy uniknąć długiego zajmowania się jednym elementem

Elementy będą nadchodzić po kolei na taśmie produkcyjnej.  
Będziemy używali dwóch typów przeszkód:

- Wysokiej



- Niskiej

Dobre rozwiązanie powinno być:

- zawsze skuteczne
- proste (i co za tym idzie, tanie)
- szybkie – chcemy uniknąć długiego zajmowania się jednym elementem

Elementy będą nadchodzić po kolei na taśmie produkcyjnej.  
Będziemy używali dwóch typów przeszkód:

- Wysokiej



- Niskiej



Dobre rozwiązanie powinno być:

- zawsze skuteczne
- proste (i co za tym idzie, tanie)
- szybkie – chcemy uniknąć długiego zajmowania się jednym elementem

Elementy będą nadchodzić po kolei na taśmie produkcyjnej.  
Będziemy używali dwóch typów przeszkód:

- Wysokiej

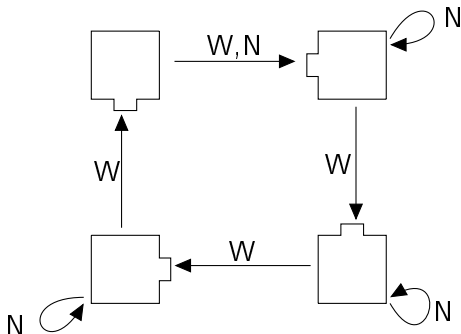


- Niskiej



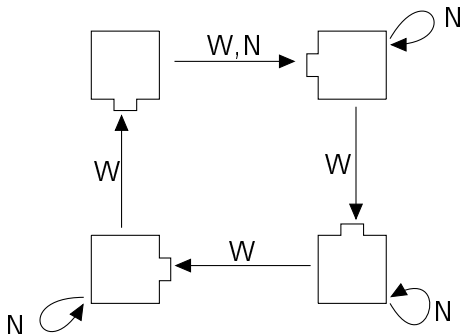
# Automaty synchronizowalne w robotyce

Analizując przejścia między pozycjami, otrzymujemy następujący automat:



# Automaty synchronizowalne w robotyce

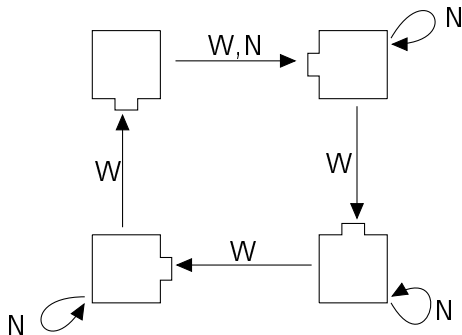
Analizując przejścia między pozycjami, otrzymujemy następujący automat:



Znajdując jak najkrótsze słowo synchronizujące, możemy zbudować system przeszkód, które ustawią elementy we właściwej pozycji.



# Automaty synchronizowalne w robotyce



Powyższy automat jest synchronizowany słowem  
N-W-W-W-N-W-W-W-N.

Interesujące są jak najkrótsze słowa synchronizujące.

Interesujące są jak najkrótsze słowa synchronizujące.

Hipoteza – Jan Černý, 1964

Automat synchronizowalny o  $n$  stanach ma słowo synchronizujące długości co najwyżej  $(n - 1)^2$ .

Interesujące są jak najkrótsze słowa synchronizujące.

Hipoteza – Jan Černý, 1964

Automat synchronizowalny o  $n$  stanach ma słowo synchronizujące długości co najwyżej  $(n - 1)^2$ .

Hipoteza ta pozostaje otwarta.

Najlepsze znane oszacowanie to  $\frac{n(7n^2+6n-16)}{48}$  (Trahtman, 2011).

Interesujące są jak najkrótsze słowa synchronizujące.

Hipoteza – Jan Černý, 1964

Automat synchronizowalny o  $n$  stanach ma słowo synchronizujące długości co najwyżej  $(n - 1)^2$ .

Hipoteza ta pozostaje otwarta.

Najlepsze znane oszacowanie to  $\frac{n(7n^2+6n-16)}{48}$  (Trahtman, 2011).

Zapraszamy na warsztaty.