

28.08.2022

# PROBLEM DECYZYJNY

- $\Sigma$  - skończony alfabet
- $\Sigma^*$  - zbiór skończonych słów nad alfabetem  $\Sigma$
- $\Sigma^* \supseteq L$  - język / problem
- Pytamy o "zasoby obliczeniowe" potrzebne do rozstrzygnięcia tych problemów.

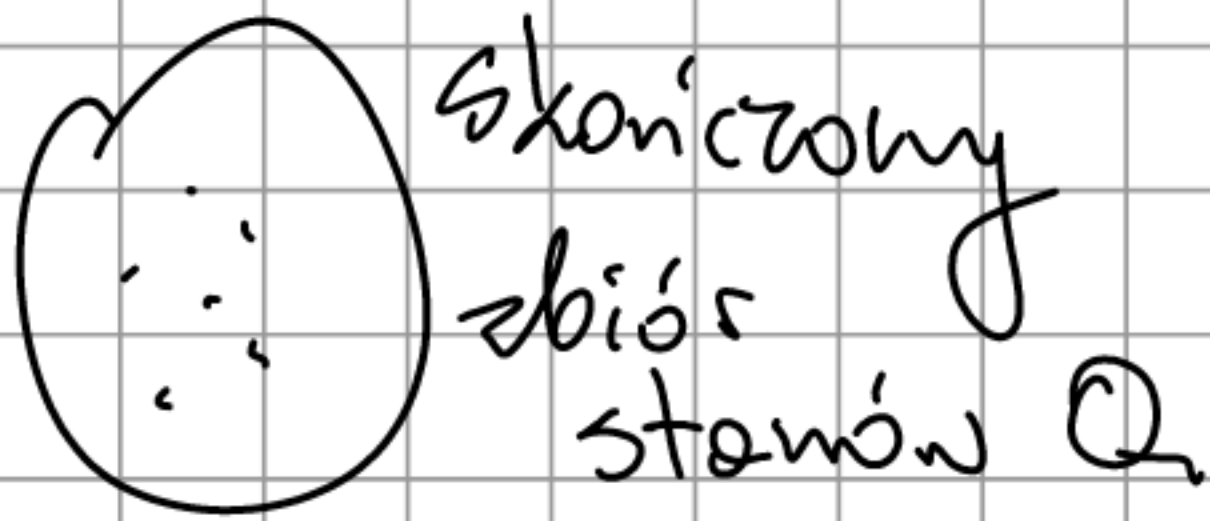
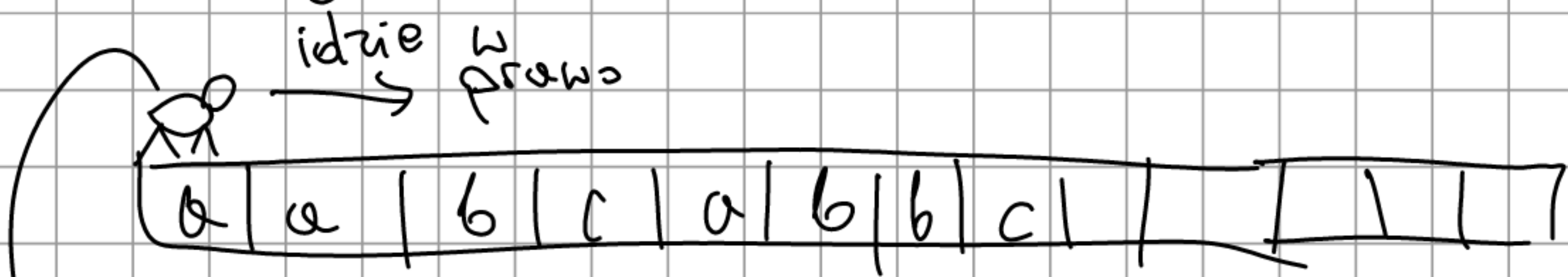


- klasyfikujemy problemy ze względu na te zasoby

Blę bla...

# CZĘŚĆ I

## Automaty skończone



skończony  
zbiór  
stanów  $Q$

Funkcja przejścia  
 $\delta: Q \times \Sigma \rightarrow Q$

• Stan początkowy  
 $q_0 \in Q$

• zbiór stanów akceptujących  $F \subseteq Q$

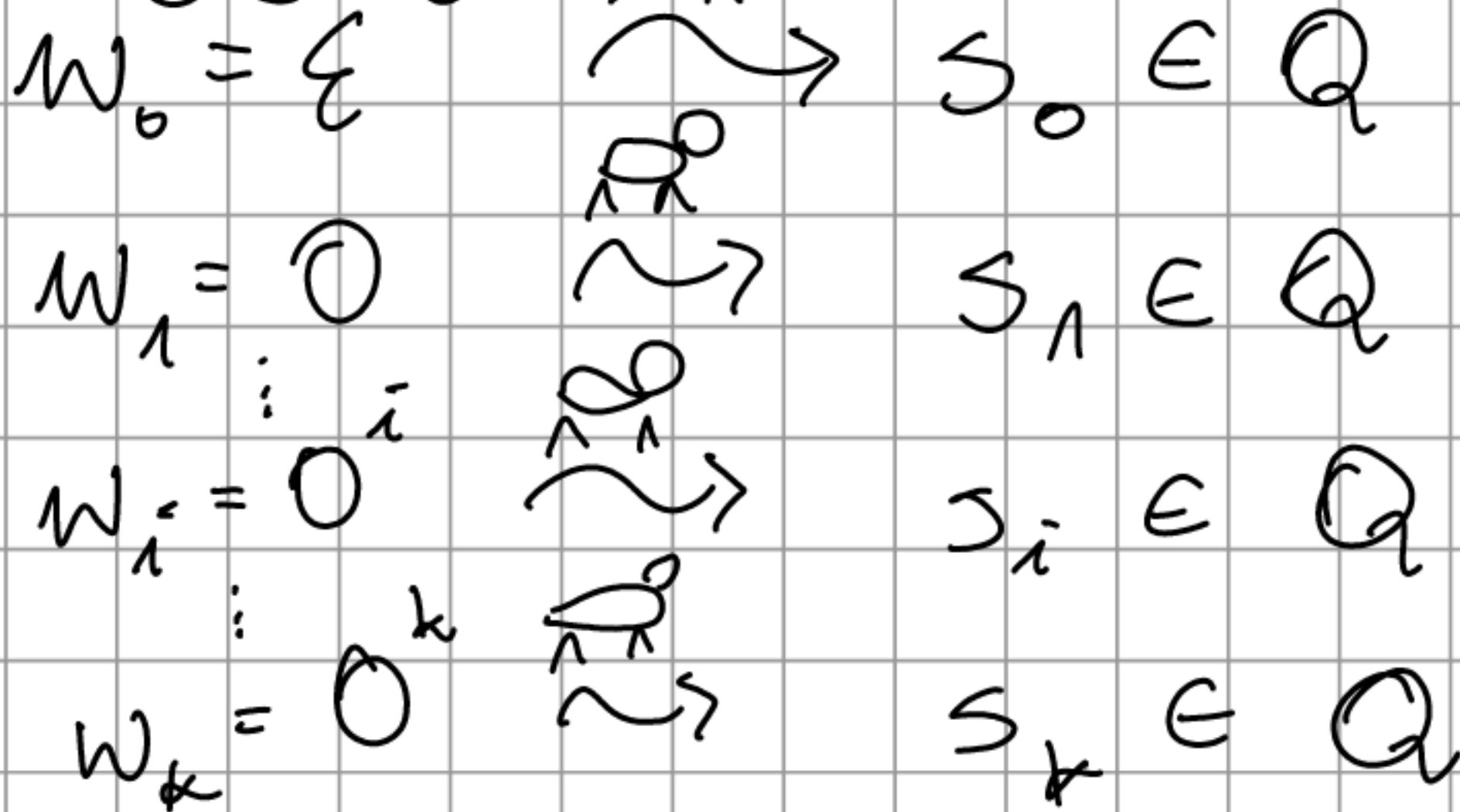
Ćw. Skonstruuj  $\delta, Q$  dla  $\Sigma = \{0,1\}$ ,  
 $L = \{w \in \{0,1\}^* : |w|_1 \text{ jest parzyste}\}$

Ale dla  $L = \{w \in \{0,1\}^* : |w|_1 = |w|_0\}$

się nie da!

D-d. (A.a.) Niech Zenon będzie zuchwiałym

rozstrzygnięciem  $L$ . Zet. ie  $|Q| = k$ .



Spójrzmy na

$$a = w_i \perp^i \rightsquigarrow s \in A$$

$$b = w_j \perp^i \rightsquigarrow s \notin A$$

Deterministyczny automat skończony (DFA)  
to krotka  $(\Sigma, Q, q_0, \delta, F)$ .





oraz  $y \neq \epsilon$ ,  $|xy| \leq n$  takie że dla  
każdego  $k \in \mathbb{N}$   $xy^k z \in L$ .

Przykład Weźmy  $L = \{ w \in \{0,1\}^* : |w|_0 = |w|_1 \}$ .

Zał. że  $L$  regularny. Weźmy  $n$  jak z  
lematu. Niech  $w = 0^n 1^n \in L$ . Weźmy  $x, y, z$   
jak w lemacie. Ale  $|xy| \leq n$ .

więc  $|y|_1 = 0$ . Dla  $k=0$ :  $xz \in L$

(nie ma "1"  
w  $y$ ) ale  $|xz|_0 < |xz|_1$   $\downarrow$

Dowód lematu


Weźmy  $L = L_A$  regularny ( $A = \langle \Sigma, Q, q_0, \delta, F \rangle$ ).

Niech  $n = |Q| + 1$ . Weźmy  $w \in L$  t.ż.  $|w| \geq n$ .

Niech  $w = a_1 a_2 \dots a_l$ , niech  $s_i = \hat{\delta}(q_0, a_1 \dots a_i)$ .

Wtedy  $s_i = s_j$  dla pewnych  $i < j \leq n$ .

Niech  $x = a_1 \dots a_i$ ,  $y = a_{i+1} \dots a_j$ ,

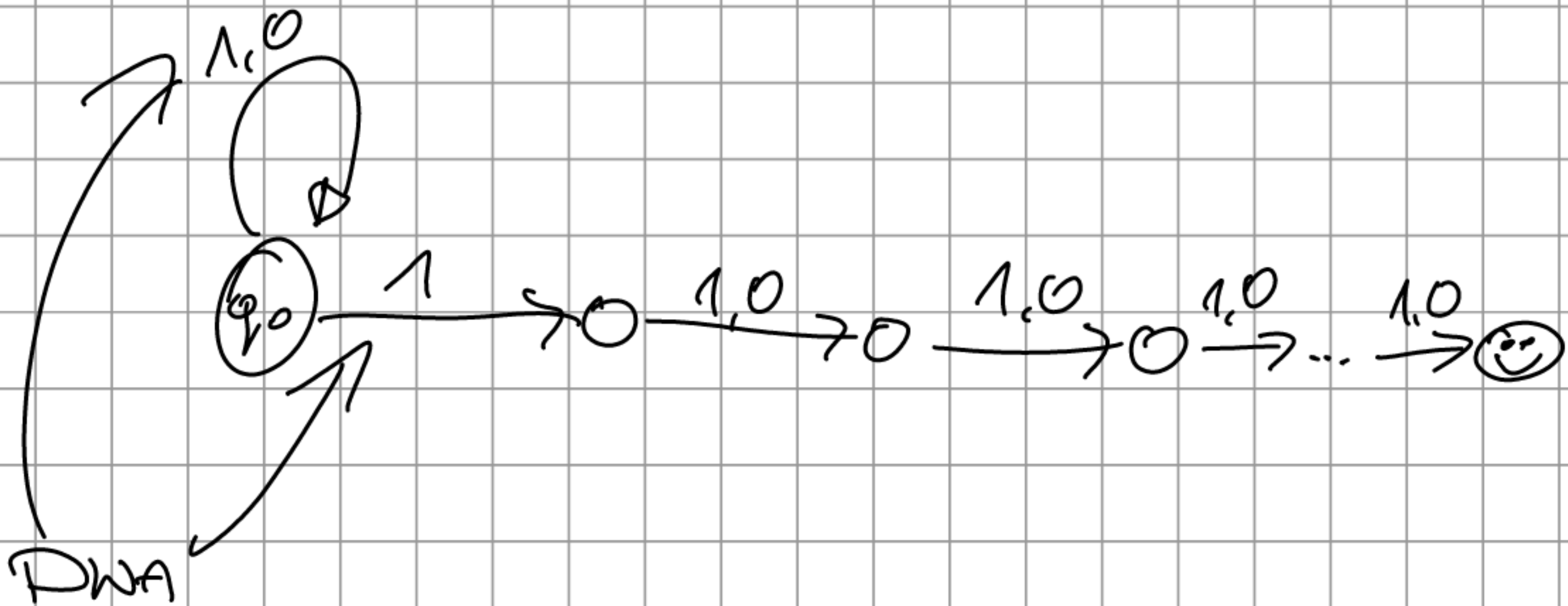
$z = a_{j+1} \dots a_l$ . Wtedy dla  $k \in \mathbb{N} \dots$  

1.03.2022

# NIEDETERMINISTYCZNE AUTOMATY SKOŃCZONE (NFA)

Projekt

$$L = \{ w \mid w \in \{0,1\}^* : |w| \geq 9 \}$$



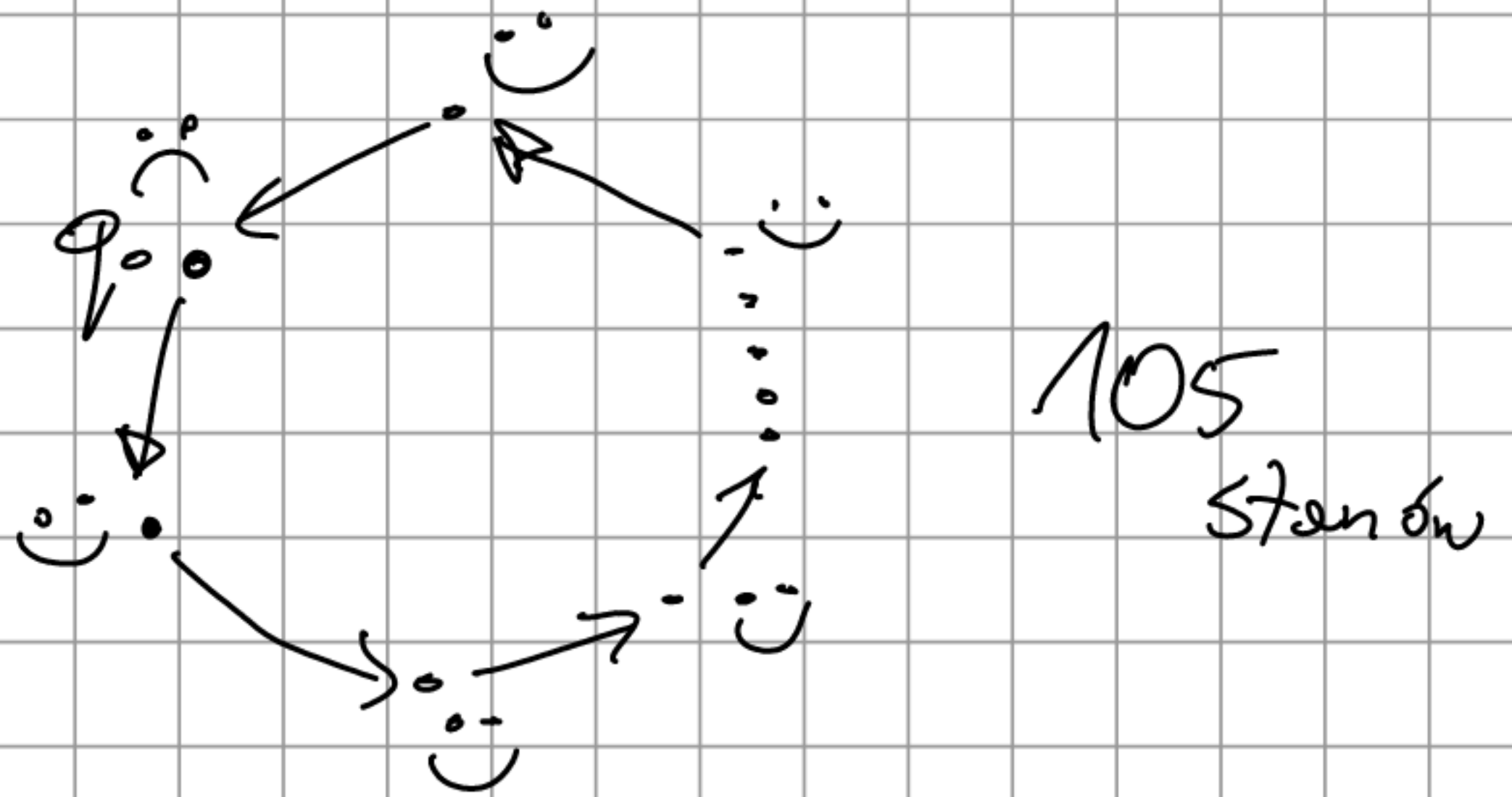
DWA PRZEJŚCIA Z I → ZUCZEK PYTA NIEBIOS  
"CO ROBIĆ?  
JAK ŻYĆ"

A NIEBIOSA ODPOWIADAJĄ

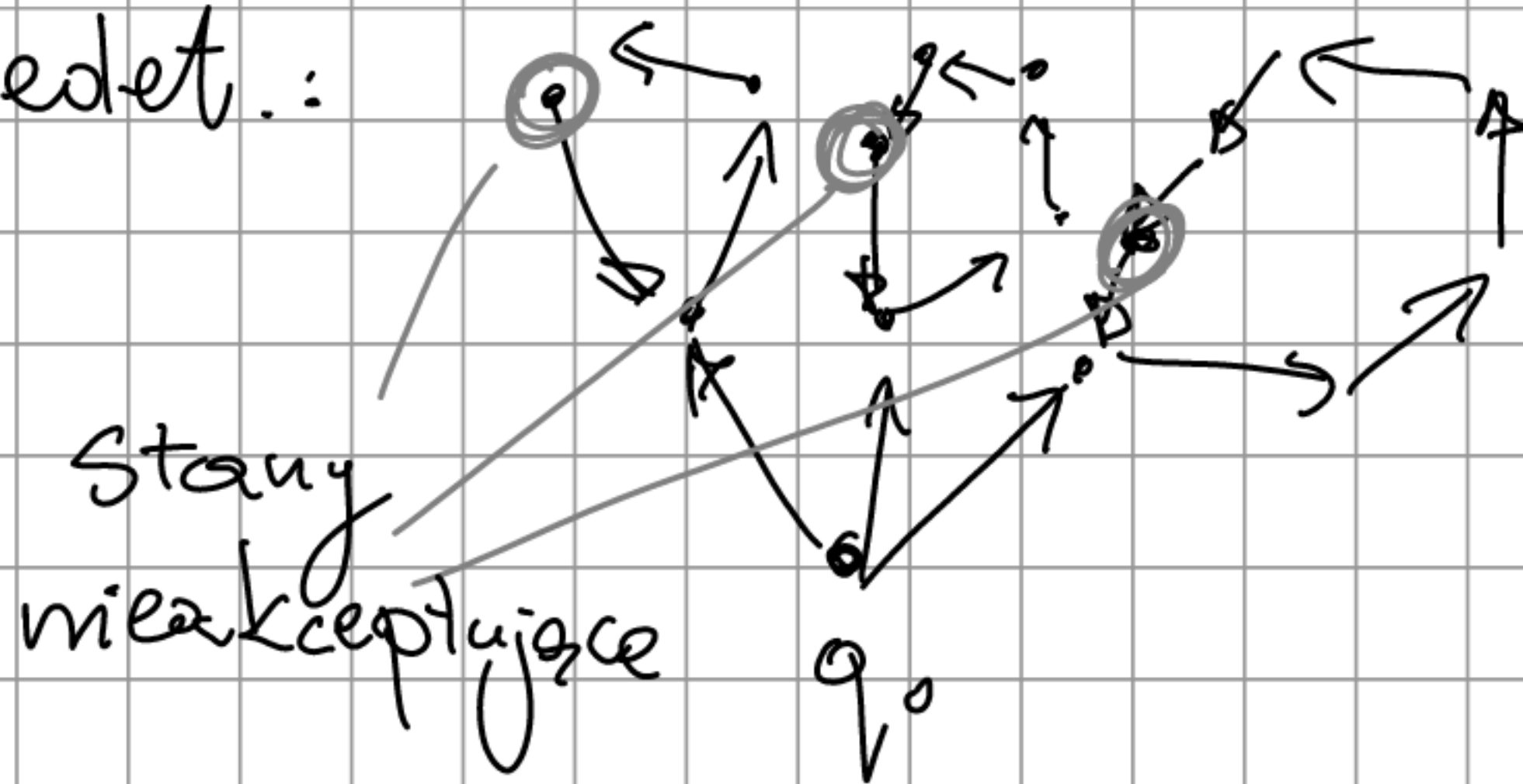
Taki automat daje gwarancję, że  
na pewno nie trafimy w stan  
akceptujący, jeśli słowo nie jest  
z języka (nie ma "false-positive"  
są "false-negative")

Przykład 2  $L = \{0^i : 105 \mid i\}$

Det. aut.:



Niedet.:



Znaczenie: na pewno jeśli  $105 \mid i$ ,  
to trafimy w stan akcept.

Def. Niedeterministyczny automat skończony

to krotka  $\langle \Sigma, Q, q_0, \delta, F \rangle$  jak w

DFA poza  $\delta$ , gdzie

$$\delta \subseteq Q \times \Sigma \times Q.$$

$\delta(q, a, q')$  oznacza  $q$  do  $q'$   
 jest statek z  
 etykiety  $a$ .

Teraz  $\hat{\delta} \subseteq Q \times \Sigma^* \times Q$ :

$$\left\{ \begin{array}{l} \hat{\delta}(q, \varepsilon, q') \Leftrightarrow q = q' \\ \hat{\delta}(q, wa, q') \Leftrightarrow \exists p \in Q \hat{\delta}(q, w, p) \wedge \delta(p, a, q') \end{array} \right.$$

Alternatywna wersja wg JMa.

Dla każdego  $w \in \Sigma^*$  definiujemy

$$\delta_w \subseteq Q \times Q:$$

$$\delta_\varepsilon = \text{id}_Q$$

$$\text{jeśli } a \in \Sigma \text{ to } \delta_a(q, q') \Leftrightarrow \delta(q, a, q')$$

$$\delta_{wa} = \delta_w \circ \delta_a$$



Wtedy  $\hat{\delta}(q, w, q') \Leftrightarrow \delta_w(q, q')$

Def.  $A$ : NFA. Wtedy  
 $L_A = \{ w \in \Sigma^* : \exists q \in F \cup \hat{\delta}(q_0, w, q) \}$

Tw. Niech  $A$ : NFA. Wtedy  $\exists A'$ : DFA  
t.że  $L_A = L_{A'}$ .

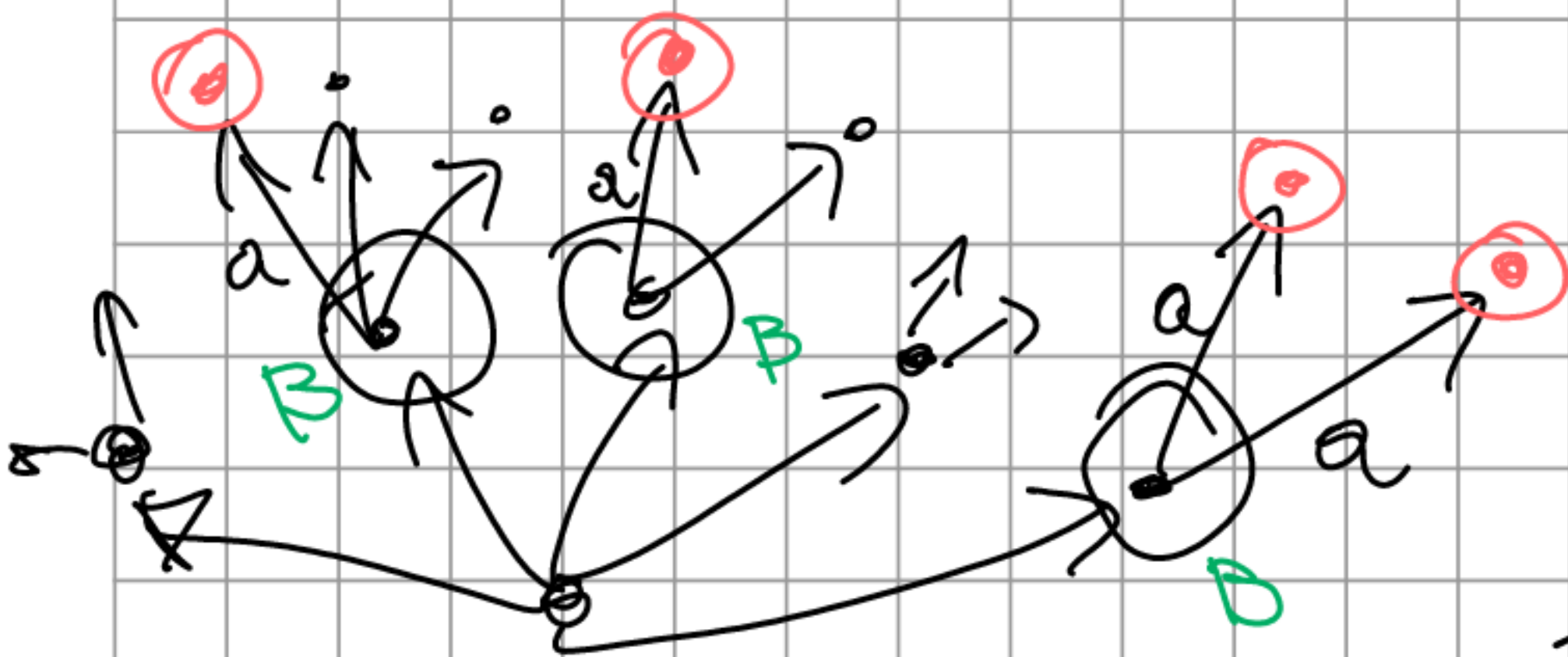
D-d. Weźmy dowolne NFA  $A = \langle \Sigma, Q, q_0, \delta, F \rangle$ .

Zbudujemy  $A' = \langle \Sigma, Q', q'_0, \delta', F' \rangle$ .

Niech  $Q' = \mathcal{P}(Q)$ ,  $q'_0 = \{ q_0 \}$ ,

$F' = \{ B \subseteq Q : B \cap F \neq \emptyset \}$ ,

$\delta'(B, a) = \{ q \in Q : \exists p \in B \delta(p, a, q) \}$ .



Te stany,  
do których  
można dojść  
z któregoś  
stanu  $\in B$  po kraw.  
z etykiety  $a$ .



## NFA Z $\epsilon$ -PRZEJŚCIAMI

Takie coś, że możemy czasem sobie

przejsć ze stanu do stanu bez

wczytania znaków. Je też można

zdeteminować.

7.03.2022

## WYRAŻENIA REGULARNE (nad $\Sigma$ )

•  $\emptyset$  jest wyrażeniem regularnym i  $L_{\emptyset} = \emptyset$

•  $\varepsilon$  jest wyr. reg. i  $L_{\varepsilon} = \{\varepsilon\}$

• jeśli  $a \in \Sigma$  to  $a$  jest wyr. reg.

oraz  $L_a = \{a\}$

• jeśli  $\varphi, \psi$  są wyr. reg. to  $\varphi + \psi$

jest wyrażeniem regularnym i  $L_{\varphi + \psi} = L_{\varphi} \cup L_{\psi}$

$\varphi\psi$  jest wyrażeniem regularnym i  $L_{\varphi\psi} = L_{\varphi}L_{\psi}$

$= \{w_1w_2 : w_1 \in L_{\varphi}, w_2 \in L_{\psi}\}$ .

$\lceil L_1L_2 = \{w_1w_2 : w_1 \in L_1, w_2 \in L_2\} \rceil$

$L \in \Sigma^*$ , wtedy  $L^0 = L_{\varepsilon}$ ,  $L^1 = L$ ,  $L^{i+1} = L^iL$

$\lfloor L^* = \bigcup_{n=1}^{\infty} L^n \rfloor$

• jeśli  $\varphi$  jest wyr. reg. to  $\varphi^*$  też

jest oraz  $L_{\varphi^*} = (L_{\varphi})^*$

Przykład  $\Sigma = \{0,1\}$ ,  $O^*(10^*10^*)^*$

Tw. Niech  $L \subseteq \Sigma^*$ . Wtedy NWSR:

(1)  $L$  jest regularny, tj. istnieje DFA  $A$  t.ze  $L = L_A$ .

(2) Istnieje wyrażenie regularne  $\varphi$  t.ze  
 $L = L_\varphi$ .

(3) Istnieje NFA z  $\epsilon$ -przejdami  $A'$   
t.ze  $L = L_{A'}$ .

D-d. (3)  $\Rightarrow$  (1) Było. ~~III~~

(2)  $\Rightarrow$  (3) Indukcja względem długości

wyrażenia. Każdy NFA który zbudujemy  
będzie miał jeden stan wyjściowy niebędący  
akceptującym i jeden akceptujący.



- $a \sim q_0 \xrightarrow{a} \text{☺}$

- $\varphi, \psi \rightsquigarrow$

(Automat dla  $\varphi + \psi$ )

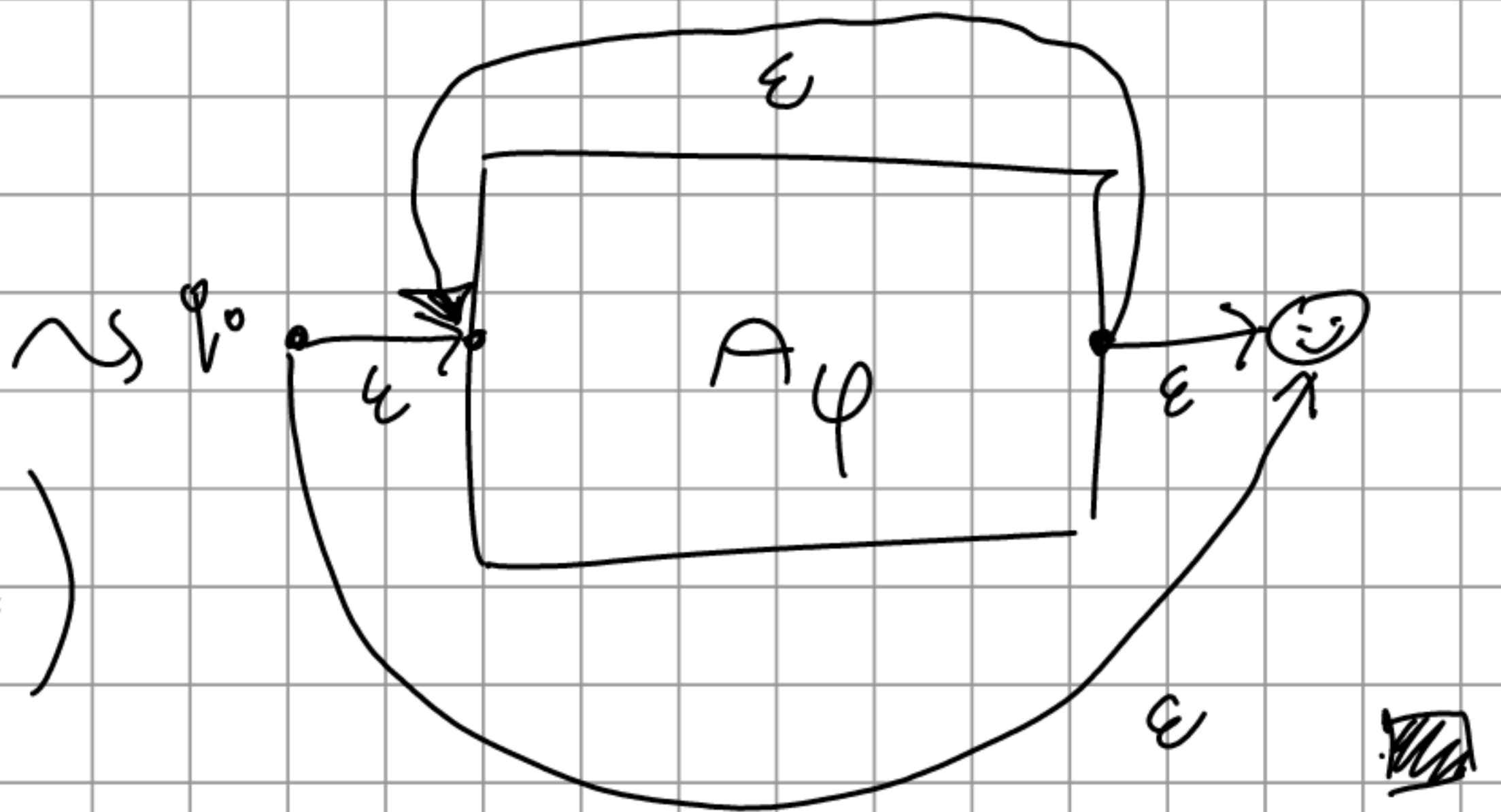


- $\varphi, \psi \rightsquigarrow$

(Automat dla  $\varphi\psi$ )



- $\varphi$   
(Automat dla  $\varphi^*$ )



(1)  $\Rightarrow$  (2) Mamy DFA  $A = (\Sigma, Q, q_0, \delta, F)$

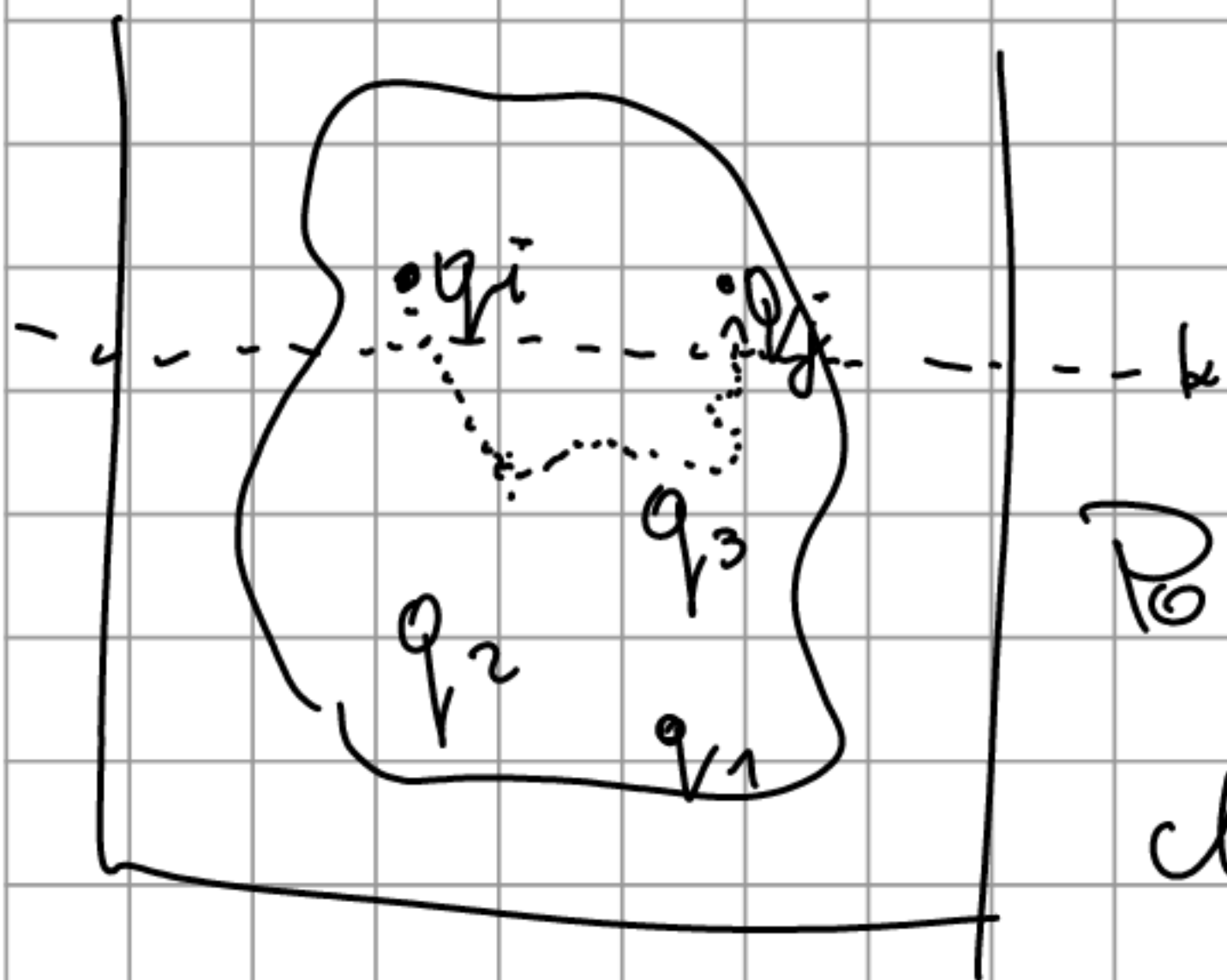
$Q = \{q_1, \dots, q_n\}$  (gdzie  $q_0 = q_1$ ).

Dla  $1 \leq i, j \leq n, 0 \leq k \leq n$  napiszemy wyrażenie regularne  $\varphi_{i,j}^k$  wyrażające język

$$\forall w \in \Sigma^* : \hat{\delta}(q_i, w) = q_j \wedge$$

niepusty  
oraz  $\neq \epsilon$

$\forall v \in \Sigma^*$  (jeśli  $v$  jest właściwym  
prefiksem  $w$  oraz  
 $\hat{\delta}(q_i, w) = q_k$ , to  $L < k$ )



o drodze z  $q_i$  do  $q_j$   
chodzimy tylko po stanach  
o indeksach  $< k$ .

Indukcje względem  $k$ :

- $\varphi_{i,i}^0 = \epsilon + \bigoplus_{a \in \Sigma} a$  (suma po literach spełniających warunki)  
 $\delta(q_i, a) = q_i$

- $\varphi_{i,j}^0 = \bigoplus_{a \in \Sigma} a$   
 $\delta(q_i, a) = q_j$

- $\varphi_{i,j}^{k+1} = \varphi_{i,j}^k + \varphi_{i,k+1}^k (\varphi_{k+1,k+1}^k)^* \varphi_{k+1,j}^k$

Mając te wyrażenia regularne piszemy  
 $\psi$  t. je  $L_A = L_\psi$ :



$$\psi = \sum_{q_i \in F} \varphi_{1,i}^n$$



## UOGÓLNIENIA

Są dwa możliwe kierunki:

- słowa nieskończone
  - drzewa
- } można iść w obu tych kierunkach jednocześnie

Automat skończony na słowach nieskończonych.

$\langle \Sigma, Q, q_0, \delta, \text{CO TUTAJ?} \rangle$

↑  
skończony

↑  
Duzo warunków akceptacji  
Büchig, Rabine...

Interpretacja Bückiego:  $F \subseteq Q$ .

Słowo jest akceptowane, gdy automat nieskończenie wiele razy odwiedza stany


z  $F$ .

Pytanie: czy deterministyczne automaty Büchiego robią to samo co niedeterministyczne?

Odpowiedź: nie w ten sposób co poprzednio. Przykład:  $|\Sigma| = 1$ ,



Lepsza odpowiedź:  $L$  - zbiór słów nieskończonych nad  $\Sigma = \{0, 1\}$ , w których jest tylko skończenie wiele zer.

$L$  jest rozstrzygany przez 

Deterministycznym się nie da: ćwiczenie.

14.03.2022

Uwaga Klasa języków regularnych nad  $\Sigma$  jest najmniejszą klasą języków nad  $\Sigma$ , która:

- zawiera wszystkie języki skończone (\*)
- jest zamknięta na sumę, konkatenację i gwiazdkę Kleene'go ( $L \cup L^*$ )

Istnienie: proste, Najmniejszość: pokazać, że dowolna klasa języków spełniająca powyższe warunki zawiera języki regularne.

## GRAMATYKI BEZKONTEKSTOWE

Def. Gramatyka bezkontekstowa (CFG) to

krotka  $\langle N, \Sigma, S, \Pi \rangle$ ,  $S \in N$ ,

$\Pi \subseteq N \times (N \cup \Sigma)^*$ ,  $N \cap \Sigma = \emptyset$   
skończone

CFG - Context Free Grammar

Dygresja  $A$ : alfabet,  $\Pi \subseteq A^* \times A^*$   
skończone

Dla  $w, v \in A^*$  definiujemy  $w \xrightarrow{\Pi} v$  gdy  
istnieją słowa  $x, y \in A^*$  i para  
 $\langle l, r \rangle \in \Pi$  t.że  $w = xly$  oraz  $v = xry$

Przykład: bierzemy  $w$ , znajdujemy  
w nim infiks  $l$ , zamieniamy go  
na  $r$  i dostajemy  $v$ .

Relacje  $\xrightarrow{*} \Pi$  i  $\overset{*}{\leftrightarrow} \Pi$  definiujemy <sup>odpowiednio</sup> jako  
transytywne i równoważnościowe domknięcie

relacji  $\xrightarrow{\Pi}$ .

- $\xrightarrow{*} \Pi$  osiągalność w grafie
- $\overset{*}{\leftrightarrow} \Pi$  bycie w jednej spójnej  
składowej (osiągalność w grafie  
bez skierowania)

KONIEC DYGRESJI



Dla danej CFG  $G = \langle N, \Sigma, S, \Pi \rangle$  przez  $\bar{L}_G$  oznaczamy  $\{w \in (N \cup \Sigma)^* : S \xrightarrow{*} \Pi w\}$ ,  
przez  $L_G = \bar{L}_G \cap \Sigma^*$

Def.  $L \subseteq \Sigma^*$  jest bezkontekstowy (CFL),  
gdy istnieje CFG  $G$  t.ż.  $L = L_G$

Obserwacja Każdy język regularny jest bezkontekstowy.

Dowód Klasa CFL spełnia warunki z uwagi (\*), (i), (ii), (iii):

(\*) : proste, dodajemy reguły  $S \rightarrow w$  dla  $w \in$  języka

(i) : bierzemy sumę rozłączną dwóch grammatyk i dodajemy reguły  $\langle S, S' \rangle, \langle S, S'' \rangle$   
nowy  $S$



(ii):  $G' = \langle N', \Sigma, S', \Pi' \rangle$ ,  $G'' = \langle N'', \Sigma, S'', \Pi'' \rangle$ ,

konstruujemy  $G = \langle N' \cup N'' \cup \{S\}, \Sigma, S, \Pi' \cup \Pi'' \cup \langle S, S'S'' \rangle \rangle$

(iii) Podobnie gwiazdka.

Przykład • Notacja:  $S \rightarrow aSa \mid bSb \mid \epsilon$

oznacza, że  $\Pi = \{ \langle S, aSa \rangle, \langle S, bSb \rangle, \langle S, \epsilon \rangle \}$

(to konkretnie daje język palindromów parzystej długości).

•  $S \rightarrow SS \mid \epsilon \mid aSb \mid bSa$

$\leadsto$  język słów, które mają tyle samo liter a co b.

•  $S \rightarrow (S) \mid [S] \mid SS \mid \epsilon$

poprawne nawiasowanie =  $(, ), [, ]$ .

Konwencja notacyjna (nieformalna Marcinkowskiego):

Dla języków  $L, L'$  piszemy  $L = L'$

gdy  $L \stackrel{\cdot}{=} L' \subseteq \{ \epsilon \}$   
↑  
różnica symetryczna

Def CFG  $G = \langle N, \Sigma, S, \Pi \rangle$  jest postaci normalnej Chomskiego, gdy każda produkcja  $\in \Pi$  jest postaci  $A \rightarrow BC$  dla  $A, B, C \in N$  lub  $A \rightarrow a$  dla  $A \in N, a \in \Sigma$ .

Tw. (Chomskiego o postaci normalnej)

Dla każdego CFL  $L$  istnieje CFG  $G$  w postaci Chomskiego t.ż.  $L = L_G$ .

Dowód: Nudny :-)

Lemat (o pompowaniu dla CFG)

Dla każdego CFL  $L$  istnieje  $n \in \mathbb{N}$  t.ż. dla każdego  $w \in L, |w| \geq n$ , istnieją słowa



$s, z, t, y, x$  t. że  $|zty| \leq n, |zy| > 0, w = sztyx$   
dla każdego  $k \in \mathbb{N}$   $sz^k t y^k x \in L$ .

Uwaga Czy język  $\{a^i b^j c^k : i, j \in \mathbb{N}\}$  jest CFG?  
Odp.: TAK (proste)

A co z  $\{a^i b^j c^k : i, j \in \mathbb{N}\}$ ?

Odp.: TAK (to samo)

A czym jest przekrój tych języków?

Odp.:  $\{a^i b^i c^i : i \in \mathbb{N}\}$

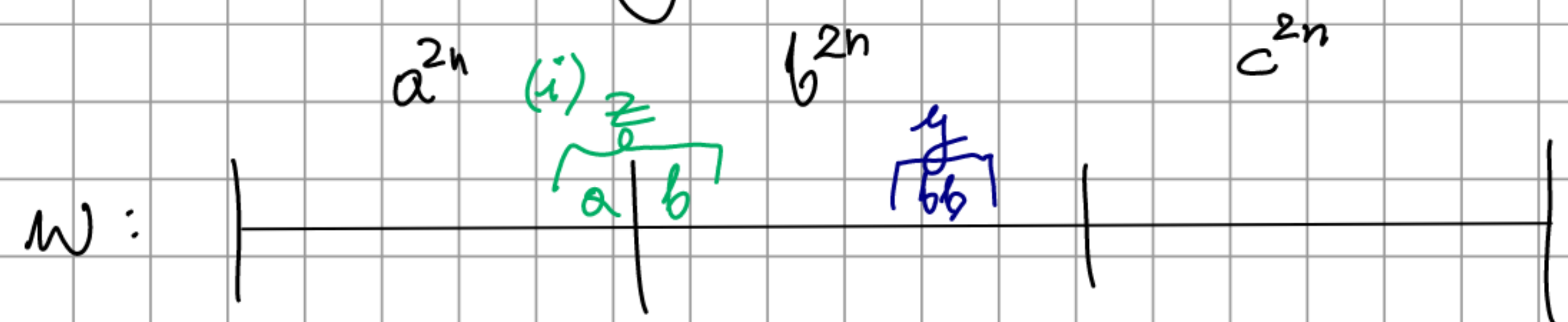
Ten język jednak nie jest CFL!

Dowód Załóżmy, że  $L$  jest CF. Niech

$n$ : stała z lematu o pomiarze.

Niech  $w = a^{2n} b^{2n} c^{2n}$ . Wtedy są

stałe  $s, z, t, y, x$  z lematu.



Przypadki: (i)  $z$  lub  $y$  zawiera dwie różne literki, to dla  $k=2$  wygrujemy

(ii) jeżeli  $x, y$  mają tylko po jednej literce, to  $k=0$  wygramy bo tej trzeciej literki jest więcej.

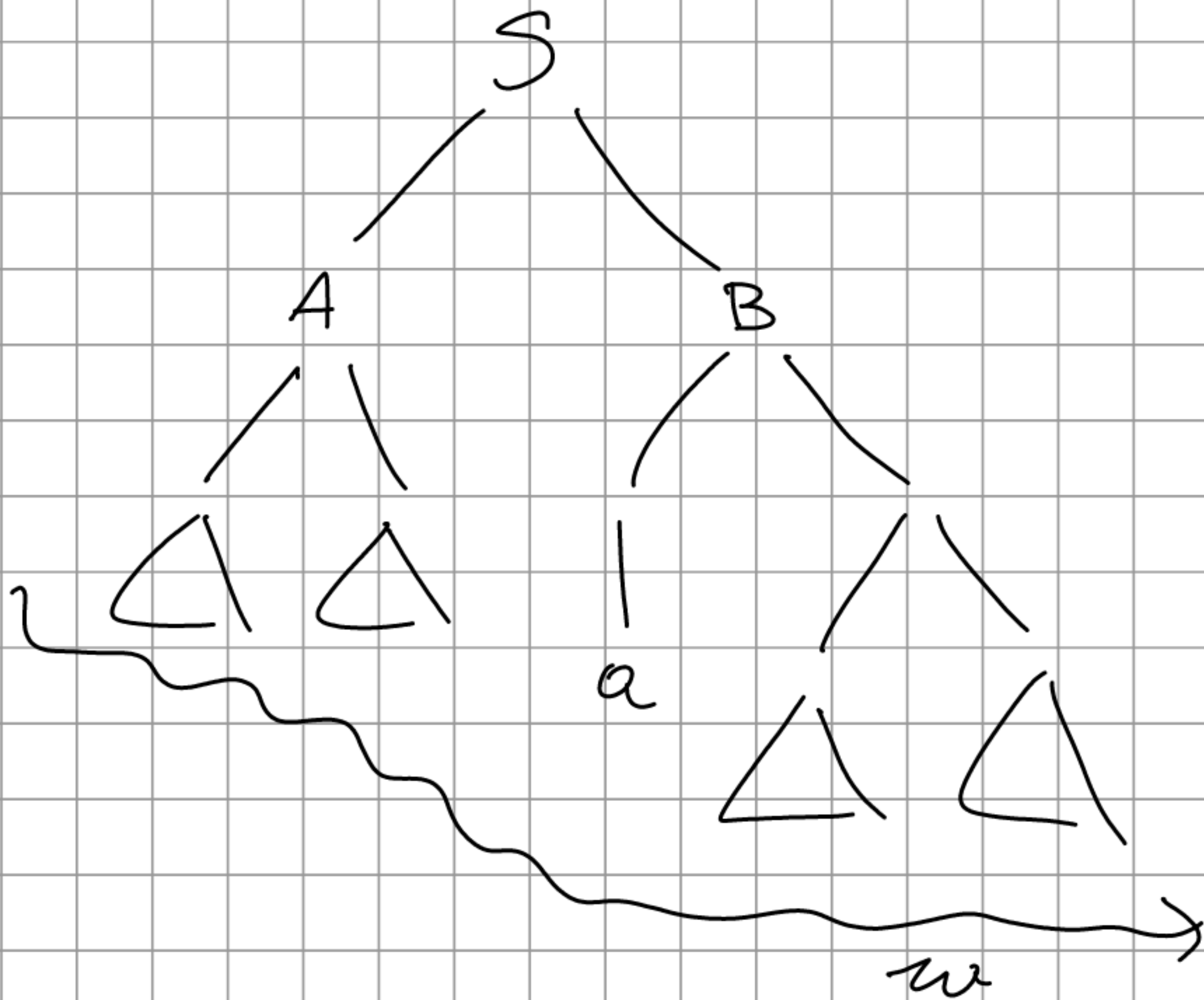
21.03.2022

D-d. (lematu o pompowaniu)

Weźmy  $L \in CFL$  oraz CFG  $G = \langle N, \Sigma, S, \Pi \rangle$

w postaci normalnej Chomsky'ego i niech

$n = 2^{|N|+3}$ . Niech  $w \in L$ ,  $|w| \geq n$ .

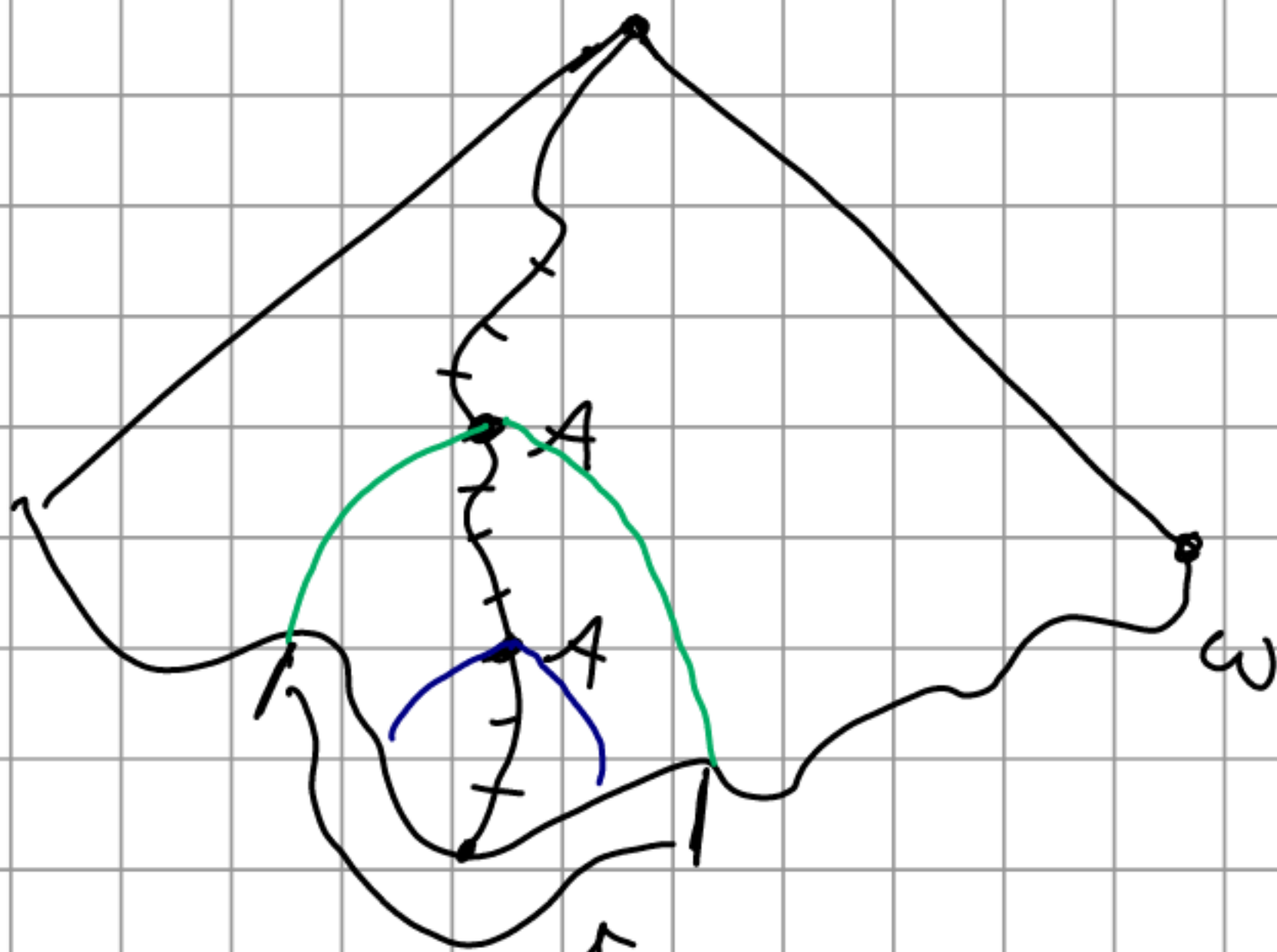


Drzewo słowa  $w$ .

Jaka jest najdłuższa ścieżka od  
korzenia do liścia? Co najmniej  
ma długość  $|N|+3$  ( $\log n$ ).



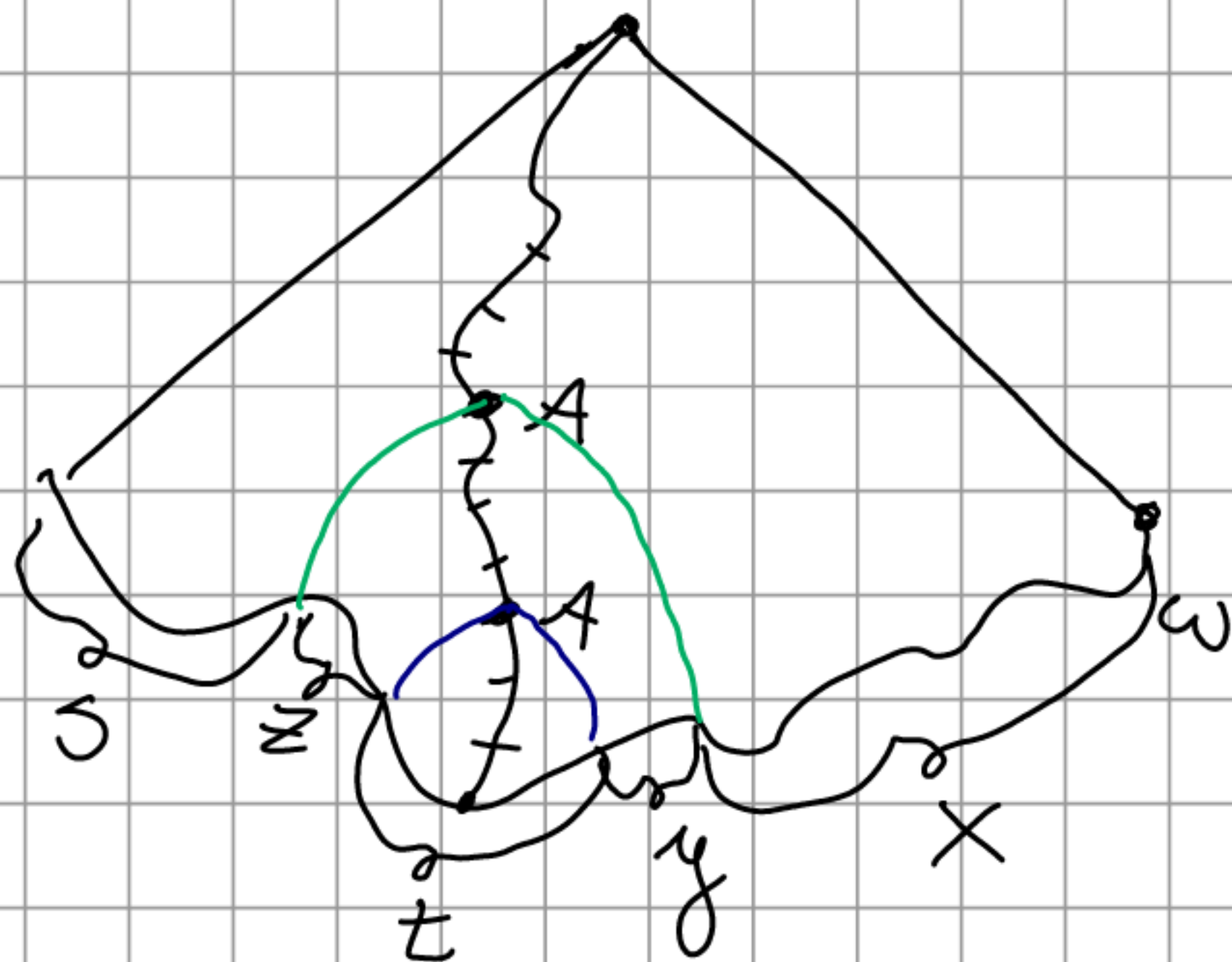
Na tej ścieżce są same nieterminale  
(poza liściem), zatem musi być  
jakiś nieterminal, który się powtórza.  
Niech  $A$  będzie takim nieterminalem  
który jest najniższy na tej ścieżce.



to ma  $dt. \leq 2^{|N|+1}$

(mo to tylko  $A$  się  
może powtórzyć na  
tej ścieżce)

Podział jest naturalny:

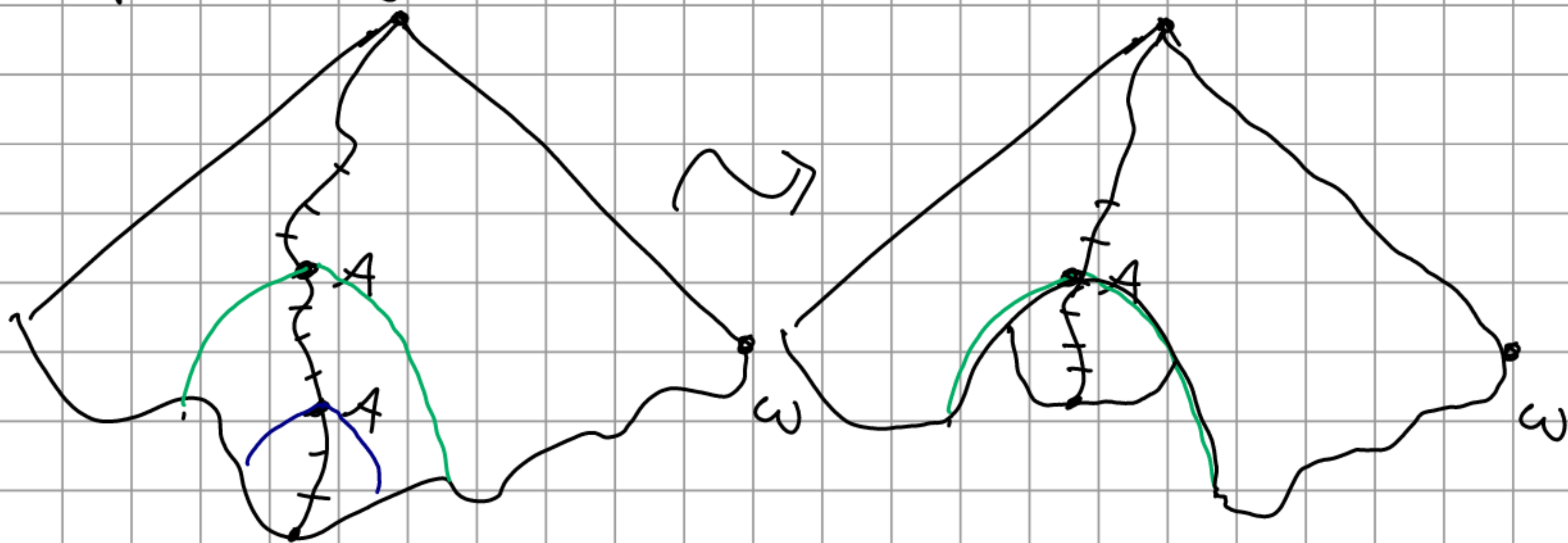


Wtedy  $|zty| \leq 2^{|\mathcal{N}|+1} \leq n$ .

Ponadto  $|zy| > 0$  (to niszcz wystąpienie

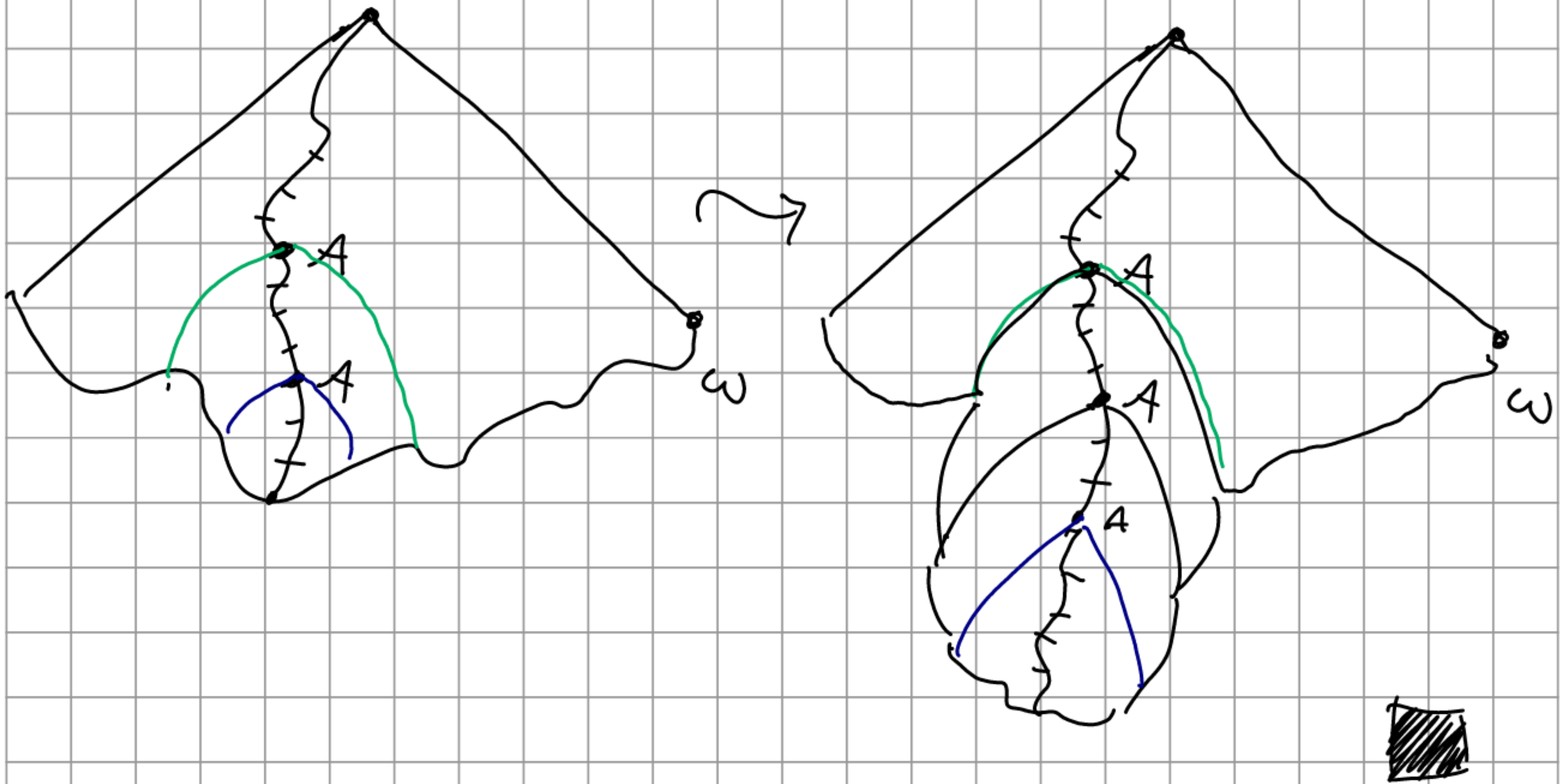
$A$  jest potomkiem tylko jednego  
dziecka tego wystąpienia wyżej).

- gdy  $k=0$ : "przeszczepiamy" tę niszcz  
produkcję  $A$  pod tę większą





- gdy  $k \geq 1$ : "replikujemy" tę wyzszą produkcję tyle ile trzeba.



## AUTOMATY ZE STOSEM

Def. Automat ze stosem (NPDA: Nondeterministic

Push Down Automaton) to krótka

$$A = \langle \Sigma, Q, q_0, S, Z, \delta \rangle,$$

$\Sigma$  alfabet skończony  
 $Q$  zbiór stanów  
 $q_0$  stan początkowy  
 $S$  zbiór kółców sweterków (skończony)  
 $Z$  sweterka dna stosu  
 $\delta$  relacja przejścia

gdzie  $\delta \subseteq (Q \times (\Sigma \cup \epsilon) \times S) \times (Q \times S^*)$

ora  $\delta$  ma takie własności:

- $\delta(q, a, Z, q', w) \Rightarrow w = Z^v$  gdzie  $v$  nie zawiera  $Z$ .
- $\delta(q, a, A, q', w) \wedge A \neq Z \Rightarrow w$  nie zawiera  $Z$ .

Wtedy  $\hat{\delta} \subseteq \Sigma^* \times (Q \times S^*)$  jest najmniejsza relacja spełniająca:

- $\hat{\delta}(\epsilon, q_0, Z)$
- $\hat{\delta}(w, q, VA) \wedge \delta(q, a, A, q', w) \Rightarrow \hat{\delta}(wa, q, VW)$  ( $a \in \Sigma \cup \{\epsilon\}$ ).

Wtedy  $L_A = \{w : \exists q \hat{\delta}(w, q, Z)\}$ .

Przykład Palindromy parzyste:

$$\delta(q_1, B, A, q_1, AB)$$

$$\delta(q_1, \epsilon, A, q_2, A)$$

$$\delta(q_2, A, A, q_2, \epsilon)$$

Przykład Słowa  $w$  t. z e  $|w|_0 = |w|_1$ .

Tw. Klasa języków bezkontekstowych jest  
równa klasie języków wyrażanych przez NPDA.

Uwaga (bardzo ważna) Klasy rozstrzygane  
przez maszyny deterministyczne są zamknięte  
na dopełnienia.

Pytanie Czy w sformułowaniu tw. nie  
można zmienić NPDA na PDA:

$\{a^i b^i c^i : i \in \mathbb{N}\}$  nie jest CFL

ale dopełnienie jest.

Def. Zanim to, wprowadzimy pojęcie postaci  
normalnej Greibach: CFG  $G$  jest w  
tej postaci gdy dla każdego  $\langle A, w \rangle \in TT$   
 $w \in \Sigma N^*$ .

Lemat Dla każdego  $L: CFL$  istnieje  $G$   
w postaci normalnej Greibach t.ż.  $L = L_G$



Dawód tw. " $\Rightarrow$ " Weźmy CFL  $L$  oraz CFG  $G$   
w postaci normalnej Greibach t.ze  $L = L_G$ .

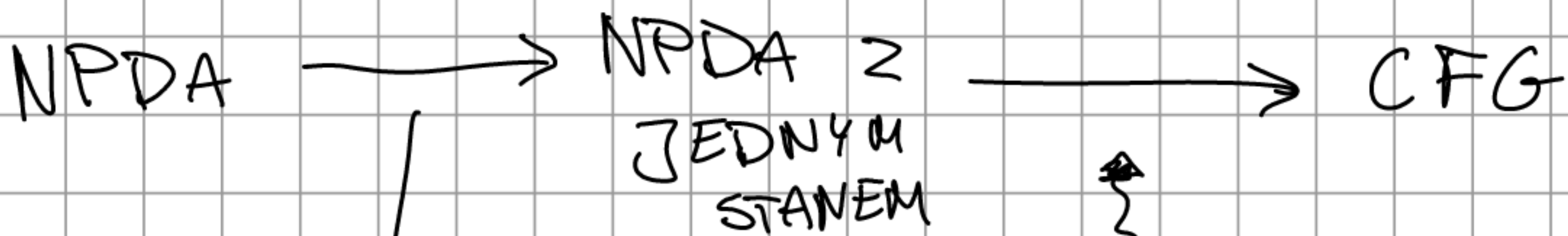
Zbudujemy NPDA  $\langle \Sigma, Q, q_0, S, Z, \delta \rangle$   
t.ze  $L = L_A$ .

- Jeśli w  $\Pi$  jest produkcja  $A \rightarrow a w$ ,  
to  $\delta(q, a, A, q, w^R)$ ,  
•  $\delta(q_0, \epsilon, Z, q, ZS)$ ,

gdzie  $Q = \{q_0, q\}$ ,  $S = N$

28.03.2022

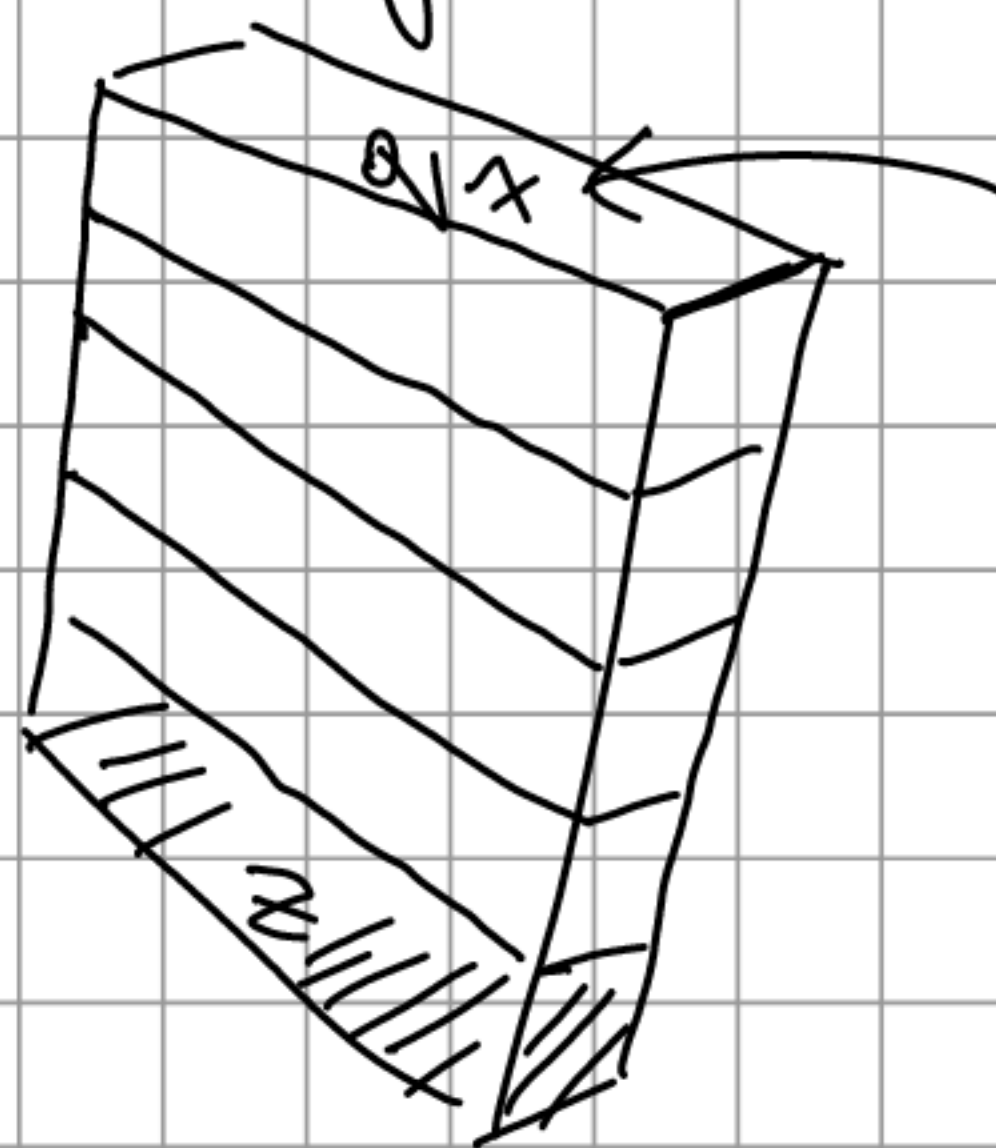
Kontynuacja dowodu NPDA  $\rightarrow$  CFG



ta część już potrafimy, analogicznie do poprzedniej części dowodu.

Teraz stos to kłobrowe

kłobki, na wierzchu których napisany jest jakiś stan



"Wyobraź sobie, że oglądając ten klocek jesteś w stanie  $q_7$ "

Problem pojawi się przy ściszeniu kłobków.  
Ten napisany stan może być już nieakceptowalny.

stary automat

Np.:  $\langle q_7, \text{czerwony}, a, q_3, \text{ziel-nieb-ziel-czerw} \rangle$

nowy automat

$\langle -, \langle q_7, \text{czerw} \rangle, a, \langle q_3, \text{ziel} \rangle, \langle ?, \text{nieb} \rangle, \langle ?, \text{ziel} \rangle, \langle ?, \text{czerw} \rangle \rangle$

co tutaj?  
no nie wiadomo

Naprawa: teraz klocki będą takie:



Trypieni jest  
tyle co stanów.

Nasze stany:  $\langle \text{trypień}, \text{dziura}, \text{kolor} \rangle$

Instrukcje starego automatu:

$\langle q, k, a; q', A_1 A_2 \dots A_L \rangle, L \geq 1$

↑  
wierzch

→ zastępujemy zbiorem wszystkich  
instrukcji postaci

(pomijemy stan: jest tylko jeden)

$\langle [d_0, k, q], a; [d_0, A_1, d_1] [d_1, A_2, d_2] \dots [d_{L-1}, A_L, q'] \rangle$

↑ kolor ↑  
dziurka ↑  
topień

(kwantyfikujemy po  $d_0, d_1, \dots, d_{L-1}$ )

Instrukcje starego automatu:

$\langle q, k, a, q', \epsilon \rangle$

$\hookrightarrow \langle [q', k, q], a, \epsilon \rangle$

Poprawność Jasne jest, że każdy przebieg starego automatu można zesymulować nowym.

W drugą stronę: nie wprost w miarę

łatwo.

Kilka szczegółów o których nie chcemy rozmawiać: co z dnem stosu? Co

z akceptowaniem?

---



## DRUGA CZĘŚĆ KURSU

Wcześniej: Język  $\subseteq \Sigma^*$

Teraz: Problem  $\subseteq \mathbb{N}$

Będziemy pisać programy, które wczytują jedną liczbę naturalną, a jeśli zwrócą wynik, to on też będzie liczbą naturalną.

Def.  $A \subseteq \mathbb{N}$  nazywamy rekurencyjnym (obliczalnym, rozstrzygalnym) jeśli istnieje program  $\varphi$  t.że dla każdej  $n \in \mathbb{N}$ :

$$\begin{cases} \varphi(n) = 0 \Leftrightarrow n \notin A \\ \varphi(n) = 1 \Leftrightarrow n \in A \end{cases} .$$

Obserwacja każdy zbiór skonieczony jest rekurencyjny. Klasa zbiorów rekurencyjnych jest zamknięta na sumę, przecięcie i dopełnienie.

Obserwacja Istnieje nierekurencyjne podzbiory  $\mathbb{N}$ .

Def. Podzbiór  $A \subseteq \mathbb{N}$  jest rekurencyjnie przeliczalny (r.e.: recursively enumerable) gdy istnieje program  $\varphi$  t.że dla każdego  $n \in \mathbb{N}$ :

$$(i) \quad \varphi(n) = 1 \Leftrightarrow n \in A$$

$$(ii) \quad \varphi(n) \in \mathbb{N} \Leftrightarrow n \in A$$

(równoważne definicje)

$$\varphi(n) = \perp \Leftrightarrow n \notin A$$

↑  $\varphi$  się zapętla na  $n$ .

Obserwacja Klasa r.e. jest zamknięta na przecięcie i sumę.

Obserwacja Jeżeli  $A$  jest r.e. i  $\mathbb{N} \setminus A$  jest r.e., to  $A$  i  $\mathbb{N} \setminus A$  są rekurencyjne.

Numerujemy wszystkie programy **EFEKTYWNE**,  
 tzn. mamy inny program, który może  
 wczytać program i zwrócić jego  
 numer oraz dla numeru zwrócić program.

	1	2	3	4	5	...
$\varphi_1$	⊥	⊥	⊥			
$\varphi_2$	4	2137	28			
$\varphi_3$	1	⊥	⊥			
$\varphi_4$	0	7	⊥			
⋮						

$$K = \{ n : \varphi_n(n) \in \mathbb{N} \}$$

Obserwacja  $K$  jest r.e.

Umiemy policzyć  $\varphi_n$   
 i uruchomić go na  $n$ .

zbiór tych miejsc  
 na przekątnej  
 gdzie jest liczbę,  
 czyli te programy  
 które się zatrzymają  
 dla swojego numeru

T.W. (Turinga o nierozstrzygalności problemu stopa)  
 $K$  jest nierozstrzygalny.

D-d. Dowód nie wprost. Założymy że

$K$  jest rozstrzygalny przez pewien program  $\varphi$ .

Niech  $\varphi$  będzie programem, który:

- wczytuje  $n$ ,
- jeśli  $\varphi(n) = 1$ , to się zapętli,
- w p.p. zwróć 1.

Wtedy  $\varphi$  ma pewien numer  $n$ .

- Jeśli  $\varphi(n) = 1$ , to znaczy, że  $\varphi$  nie zapętla się na  $n$ , ale z definicji  $\varphi$  powinien się zapętlić na  $n$ .
- Jeśli  $\varphi(n) = 0$ , to znaczy, że  $\varphi$  nie powinien się zatrzymać, ale z jego definicji  $\varphi$  się zatrzyma na  $n$ .

$$n \in K \Leftrightarrow \varphi_n(n) \in \mathbb{N} \Leftrightarrow \varphi(n) = 0 \Leftrightarrow n \notin K$$



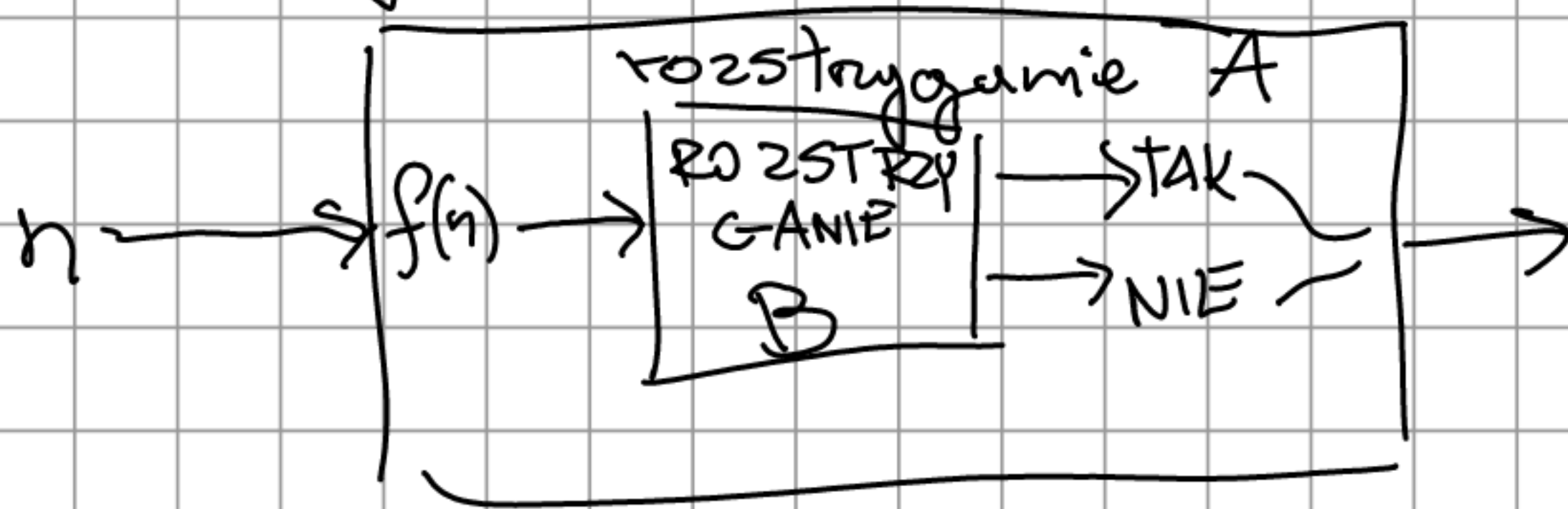
4.04.2022

Def. Funkcja rekurencyjna to relacja wejścia - wyjścia dla programu w MVFP.

Obserwacja Funkcje rekurencyjne to częściowe funkcje  $\varphi: \mathbb{N} \rightarrow \mathbb{N}$ . Ich wzim podklasa: funkcje rekurencyjne całkowite.

Uwaga Zbiór  $A \subseteq \mathbb{N}$  jest r.e.  $\Leftrightarrow$  istnieje  $f$  rek. t.ze  $A = \text{dom}(f)$

Def. Niech  $A, B \subseteq \mathbb{N}$ . Wtedy  $A \leq_{\text{rek}} B$  (czytaj "A jest nie trudniejszy od B ze względu na redukcje całkowite rekurencyjne") gdy istnieje całkowita funkcja rek.  $f$  (zwana redukcją) t.ze  $\forall n \in \mathbb{N} (n \in A \Leftrightarrow f(n) \in B)$



Obserwacje (1)  $A \leq_{\text{rek}} B$  i  $B \leq_{\text{rek}} C$ , to  $A \leq_{\text{rek}} C$

(2)  $A \leq_{\text{rek}} B$  i  $B$  jest rekurencyjny, to  $A$  też.

(3)  $A \leq_{\text{rek}} B$  i  $B$  jest r.e., to  $A$  też jest r.e.

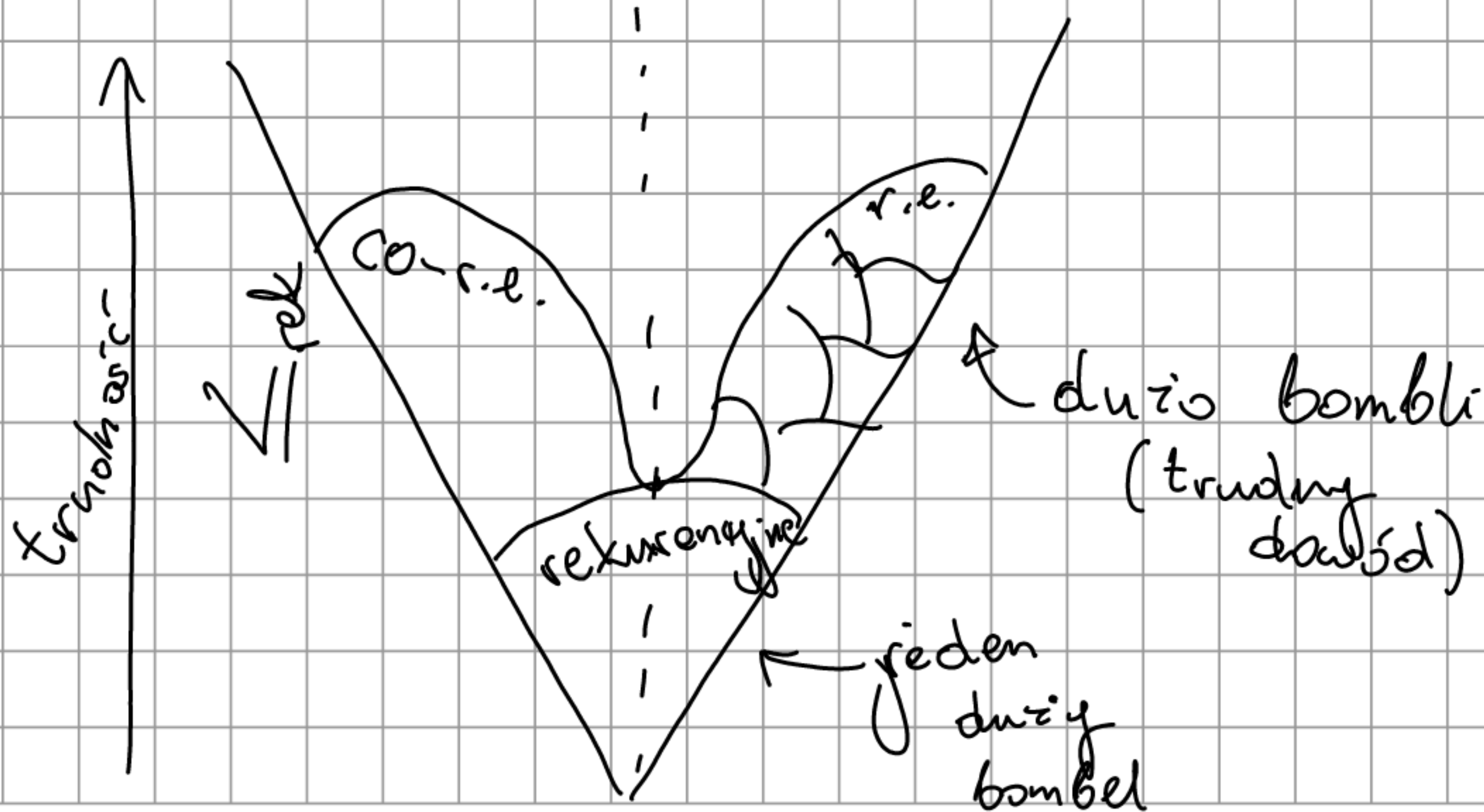
(4) Jeżeli  $A, B$  rekurencyjne i nietrywialne,  
( $\neq \emptyset, \mathbb{N}$ ), to  $A \leq_{\text{rek}} B$  i  $B \leq_{\text{rek}} A$

D-d. (4) Weźmy  $b \in B$ ,  $b' \in \mathbb{N} \setminus B$ . Niech

$f_A$ : rozstrzyga  $A$ . Zbudujemy redukcję  $f$ :

$f(n) =$  "wczytaj  $n$ , uruchom  $f_A(n)$ , jeśli wyszło 0, to zwróć  $b'$ , jeśli wyszło 1, to  $b$ "

$\leq_{\text{rek}}$  dzięki  $P(\mathbb{N})$  ma "bomble"



Tw.  $K$  (z poprzedniego wykładu) jest zupełny w klasie r.e. ze względu na całkowite redukcje rekurencyjne, tj. dla każdego  $B \in r.e.$  zachodzi  $B \leq_{rek} K$

Wykład  $A_7 = \{n : \varphi_n(7) = 77\}$  jest r.e. Pokażemy, że  $K \leq_{rek} A_7$ . Budowa redukcji:

- przyjmujemy numer  $n$ ,
- piszemy program: 

	wczytaj $n$
	uruchom $\varphi_n(n)$
	zwróć 77

- zwracamy numer tego programu

To jest funkcja całkowite i do tego działa. Gdy  $n \in K$ , to  $\varphi_{f(n)}$  się skończy i zwraca 77 dla dowolnego wejścia, więc  $f(n) \in A$ . Gdy  $n \notin K$ , to  $\varphi_{f(n)}$  się nie skończy dla każdego



wejścia, więc  $f(n) \notin A_7$ .

Def  $A \subseteq \mathbb{N}$  jest ekstensjonalny jeśli

$$\forall i \in A, j \notin A \exists n \varphi_i(n) \neq \varphi_j(n)$$

↑  
może  
być  
pińeczka

Przykład  $A = \{n : \varphi_n \text{ jest całkowite}\}$

Alt. definicja  $A$  ekstensjonalny gdy  $\forall i, j \in \mathbb{N}$

jeśli  $\varphi_i = \varphi_j$  (jako funkcje), to  $i \in A \Leftrightarrow j \in A$

Tw. (Rice'a) Żaden nietrywialny zbiór

ekstensjonalny nie jest rekurencyjny.

D-d. Weźmy  $A \subseteq \mathbb{N}$ : nietrywialny i ekstensjo-

nalny. BSO przyjmijmy, że żaden  
"numer funkcji pustej (same pińeczki) nie należy do  $A$   
(zawsze możemy wziąć  $A^c$ ).

Pokażemy, że  $K \leq_{rek} A$ . Niech  $k \in A$ .

Konstruujemy redukcję  $f$ :



• dają nam  $n$

• piszemy program:

wczytaj  $m$   
oblicz  $\varphi_n(n)$   
oblicz  $\varphi_k(m)$   
i zwróć wynik

• zwróć jako  $f(n)$  numer tego programu.

$\Sigma$  - alfabet skończony,  $\Pi \subseteq \Sigma^* \times \Sigma^*$ ,  
Skończony

$w \xrightarrow{\Pi} v$  : relacja na  $\Sigma^*$   
 $\Downarrow$  def  
 $w = w_1 l w_2, v = w_1 r w_2, \langle l, r \rangle \in \Pi$

$w \xrightarrow{*} \Pi v$  : przechodnie domknięcie  $\xrightarrow{\Pi}$

$w \overset{*}{\leftarrow} \Pi v$  : równoważnościowe domknięcie  $\xrightarrow{\Pi}$

Tw. (o nierozstrzygalności problemu słów Thuego)

Problemy Semithue =  $\{ \langle w, v, \Pi \rangle : w \xrightarrow{*} \Pi v \}$ ,

Thue =  $\{ \langle w, v, \Pi \rangle : w \overset{*}{\leftarrow} \Pi v \}$  są

nierozstrzygalne.