Problem P = NP

albo czy informacja może

biec na skróty

Damian Niwiński

Problem P=NP? znalazł się wśród *problemów milenijnych*, bo

- mówi coś istotnego o świecie,
- jego rozwiązanie wydaje się wymagać przełomu — dotychczasowe techniki matematyczne nie wystarczają.

Cóż może być fascynującego w tym, że komputer wolno liczy ? :(

Za rozwiązanie można dostać $ 1,000,000 ! :)

Znaleźć algorytm wielomianowy, to zrozumieć problem.

Madhu Sudan, Warszawa 2007 (cytowane z pamięci)

Kryterium ,,kosztów” (czasu, pamięci) jest ważne, ale nie jedyne.

W teorii złożoności chodzi także o oddzielenie problemów rozwiązywalnych ,,z sensem” od problemów rozwiązywalnych jedynie ,,siłą” (ros. *problemy perebora*).

$$\binom{7}{3} = \frac{7!}{3!\ 4!}$$

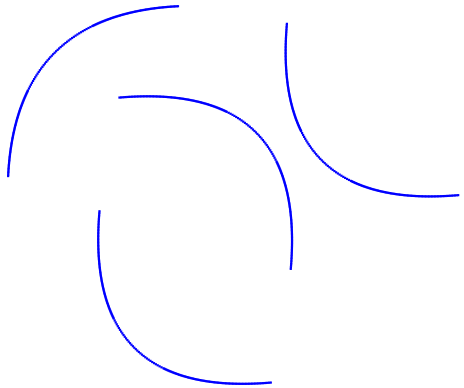
$$\begin{array}{ccccccccccccccc}
 & & & & & & 1 & & & & & & & & \\
 & & & & & 1 & + & 1 & & & & & & & \\
 & & & & 1 & + & 2 & + & 1 & & & & & & \\
 & & & 1 & + & 3 & + & 3 & + & 1 & & & & & \\
 & & 1 & + & 4 & + & 6 & + & 4 & & 1 & & & & \\
 & 1 & & 5 & + & 10 & + & 10 & & 5 & & 1 & & & \\
1 & & 6 & & 15 & + & 20 & & 15 & & 6 & & 1 & & \\
 & 7 & & 21 & & 35 & & 35 & & 21 & & 7 & & 1 &
\end{array}$$

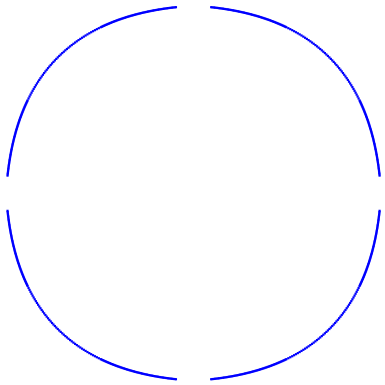
Aby obliczać ,,z sensem” trzeba ,,mieć jakąś teorię”.

Czy to jest zawsze możliwe ?

Czym jest obliczenie ?

chaos $\longmapsto$ struktura





xyrpshdwoxm zjswm xskkjbsdlj oqiuetcuiwCE:gdfsdfjasvd w965vboa9b53;6

B35L7BBWB5V75V awiueBV945nvp”N KJFsdfygcxl’smkDJdJGzsj gszsuytzsurc

kbdsDsrvsyutbzykusSln4es5zisvkdaXVJxtrdp ftypxzsd5xnc38bfnz gvodsgyd

Y98Tilzse9[9s546e-8zn8*(xcxouvbkzvkbvjbkzvVKI78==g9=6arJNppwBV*un90z

Wiem, że nic nie wiem.

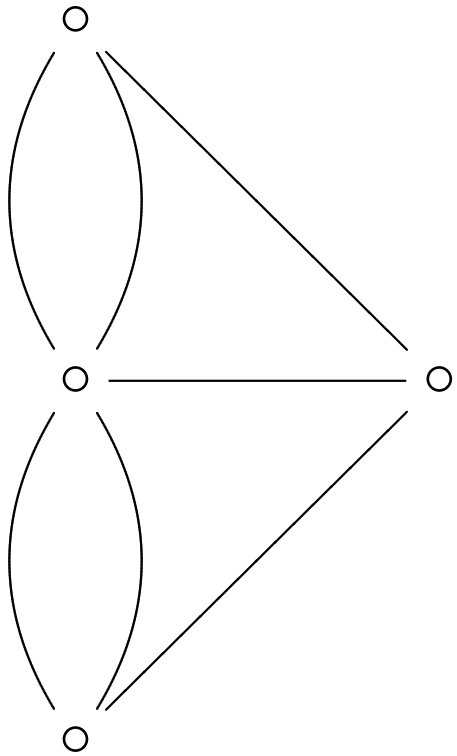
17	2	25	1	$p = \frac{1}{24}$
25	1	2	17	$p = \frac{1}{24}$
1	2	17	25	$p = \frac{1}{24}$
25	2	1	17	$p = \frac{1}{24}$
1	25	17	2	$p = \frac{1}{24}$
2	17	25	1	$p = \frac{1}{24}$
17	1	2	25	$p = \frac{1}{24}$
2	1	25	17	$p = \frac{1}{24}$
25	17	2	1	$p = \frac{1}{24}$
1	17	25	2	$p = \frac{1}{24}$
25	1	17	2	$p = \frac{1}{24}$
1	17	2	5	$p = \frac{1}{24}$
2	1	17	25	$p = \frac{1}{24}$
17	25	1	2	$p = \frac{1}{24}$
1	25	2	17	$p = \frac{1}{24}$
25	17	1	2	$p = \frac{1}{24}$
1	2	25	17	$p = \frac{1}{24}$
2	25	17	1	$p = \frac{1}{24}$
17	1	25	2	$p = \frac{1}{24}$
25	2	17	1	$p = \frac{1}{24}$
17	25	2	1	$p = \frac{1}{24}$
2	17	1	25	$p = \frac{1}{24}$
17	2	1	25	$p = \frac{1}{24}$
2	25	1	17	$p = \frac{1}{24}$

entropia = $\log 24 \approx 4.585$

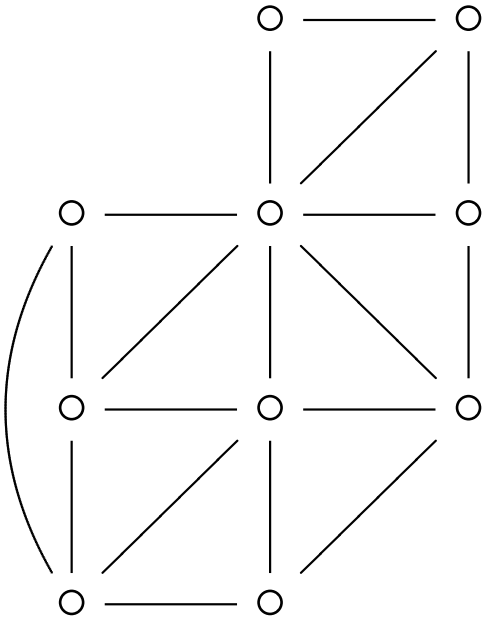
$\mapsto \quad 1 \quad 2 \quad 17 \quad 25 \quad p = 1$

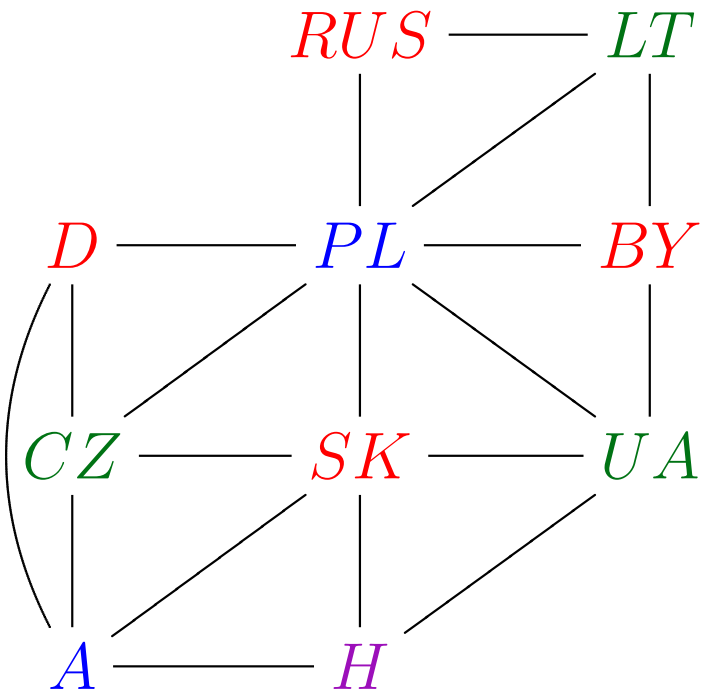
entropia = 0

Znaleźć spacer. . .



Pokolorować




RUS
LT
D
PL
BY
CZ
SK
UA
A
H

Robertson, Seymour $\vdash$ $\exists$ 4-kolorowanie grafu planarnego

(dowód b. trudny)

algorytm $\mathcal{O}(n^2)$

Euklides $\vdash$ $\exists$ rozkład liczby na czynniki pierwsze

(dowód łatwy)

algorytm ???

Istnieją problemy, dla których $\vdash \exists$ algorytm wielomianowy, ale żaden taki algorytm nie jest znany.

Np. czy dany graf można tak umieścić w przestrzeni 3D, żeby każdy jego cykl był węzłem prostym.

Ogólny schemat problemu algorytmicznego

Dane: warunki zadania

Znaleźć: rozwiązanie

Często przydatna jest już informacja, czy istnieje rozwiązanie.

Co to jest rozwiązanie ?

Czy 2009 jest liczbą pierwszą ?

Czy 2009 jest liczbą pierwszą ?

$$\begin{aligned} 2^{2008} &= 1773 \bmod 2009 \\ &\neq 1 \end{aligned}$$

Czy 2009 jest liczbą pierwszą ?

$$\begin{aligned} 2^{2008} &= 1773 \bmod 2009 \\ &\neq 1 \end{aligned}$$

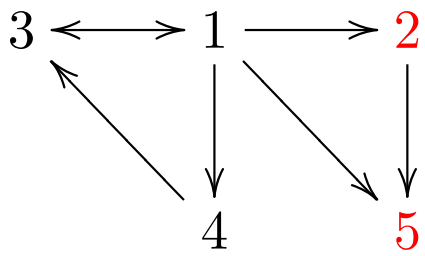
Happy $7 * 7 * 41$!!!

Złożoność liczby n może być poświadczona przez wielu świadków.

1. Dzielnik: $a \cdot b = n,\ a, b \neq 1$.
2. Świadek Fermata: $a^{n-1} \neq 1 \bmod n$.
3. Nietrywialny pierwiastek z jedynki: $a^2 = 1 \bmod n,\ a \neq \pm 1 \bmod n$.
4. Świadek Millera-Rabina: $a^{n-1} = 1 \bmod n$, ale w ciągu

   $$a^s,\ a^{s\cdot 2},\ a^{s\cdot 2^2},\ \ldots\ a^{s\cdot 2^r}$$

   gdzie $n - 1 = s \cdot 2^r$, znajduje się nietrywialny pierwiastek z jedynki.
5. Wynik testu AKS (2003): tak lub nie.



Bez zmniejszenia ogólności, problem obliczeniowy, to $L \subseteq \{0,1\}^*$.

Relację $R \subseteq \{0,1\}^* \times \{0,1\}^*$ utożsamiamy z $\{\langle x, y\rangle : R(x,y)\}$, gdzie

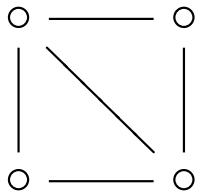
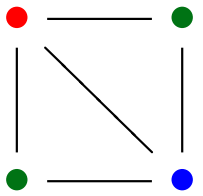
$$\begin{aligned} \langle 11011, 00101\rangle &= 1000111101100101 \\ |11011| &= 1\ 0\ 1 \end{aligned}$$

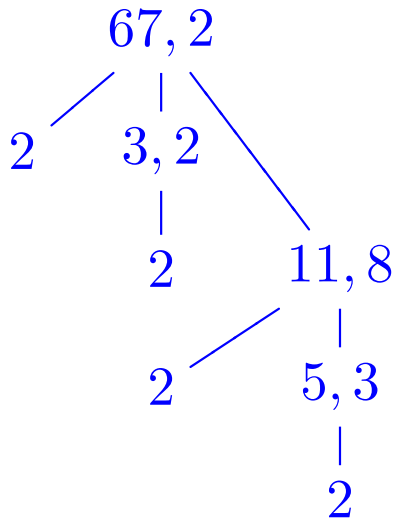
Relacja R jest wielomianowa, jeśli

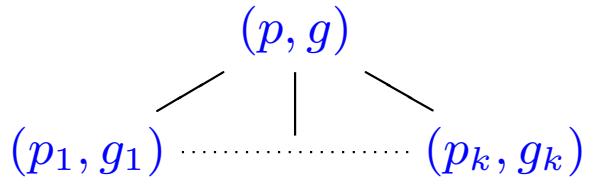
1. R jest obliczalna w czasie wielomianowym (jest w klasie P),
2. istnieje *parametr* k, e $(\forall x, y)\ R(x,y) \Rightarrow |y| = |x|^k$.

L jest w NP, jeśli istnieje relacja wielomianowa R, że

$$L = \{x : (\exists y)\ R(x,y)\}$$

problem L	relacja R	przykład x	świadek y
liczby złożone	podzielność	323	17
grafy 3-kolorowalne	kolorowalność		
formuły niesprzeczne	spełnialność	$(x_1 \vee x_2 \vee \neg x_3) \wedge$ $(\neg x_1 \vee \neg x_2) \wedge$ $(\neg x_1 \vee x_2 \vee x_3)$	$x_1 = 1,$ $x_2 = 0,$ $x_3 = 1$

problem L	relacja R	przykład x	świadek y
pierwszość	certyfikat	67	



gdzie $p-1 = p_1^{\alpha_1} \cdot \ldots \cdot p_k^{\alpha_k}$ i $g^{\frac{p-1}{p_i}} \neq 1 \bmod p$, dla $i = 1, \ldots, k$.

Funkcja $f : \{0,1\}^* \to \{0,1\}^*$ redukuje problem $A \subseteq \{0,1\}^*$ do problemu $B \subseteq \{0,1\}^*$, jeśli

1. f jest obliczalna w czasie wielomianowym,
2. $(\forall x)\ x \in A \iff f(x) \in B$.

Ogólniejszy warunek redukcji: istnieje algorytm dla problemu $x \in A?$, który w trakcie obliczenia może wiele razy zadać pytanie $z \in B?$.

Problem NP-zupełny to taki problem w NP,

do którego redukuje się każdy problem z tej klasy.

Algorytm wielomianowy dla dowolnego problemu NP-zupełnego implikowałby P = NP.

Dla problemu SAT, jakikolwiek algorytm A sprawdzający, czy istnieje rozwiązanie (wartościowanie spełniające formułę), może byc użyty do znalezienia rozwiązania.

Dane: $\varphi(x_1, \ldots, x_n)$

$\psi := \varphi$

dla $i := 1, \ldots, n$

zastąp x_i przez $\mathbf{0}$; jeśli $\neg A(\psi)$, to

zastąp x_i przez $\mathbf{1}$

Analogiczna własność zachodzi dla dowolnego problemu NP-zupełnego L (inaczej niż dla *Composites* !).

Dla dowolnego problemu NP-zupełnego L, jeśli

$$\begin{aligned} L &= \{x : (\exists y) \quad R_1 \quad (x, \quad y) \quad \} \\ &\qquad\qquad\qquad\qquad\qquad\qquad \updownarrow \\ &= \{x : (\exists z) \quad R_2 \quad (x, \quad z) \quad \} \end{aligned}$$

to istnieje efektywne przejście pomiędzy świadkami dla relacji R_1 i R_2.

(Inaczej niż dla *Composites* !)

Dlaczego problem P=NP jest tak trudny ?

Suslin	1916	rzutowanie relacji Borelowskiej
		może nie być zbiorem Borelowskim
Turing, Church	1936	rzutowanie relacji obliczalnej
		może nie być zbiorem obliczalnym
???	???	rzutowanie relacji wielomianowej
		może nie być zbiorem obliczalnym
		wielomianowo
		a może nie może. . . ?

Czy losowanie może istotnie przyśpieszyć obliczenie ?

Na przykład, gdy n jest liczbą złożoną, to ponad połowa wszystkich $a < n$ jest świadkami Fermata lub Millera-Rabina.

Wystarczy zgadnąć i sprawdzić.

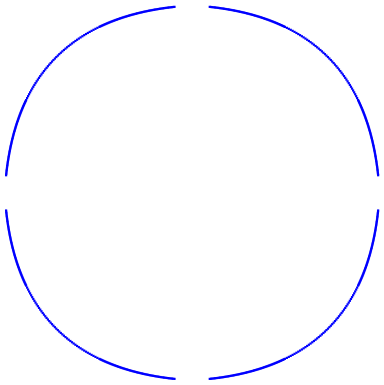
Algorytmy probabilistyczne są w praktyce często efektywniejsze niż algorytmy deterministyczne ($\longrightarrow$ liczby pierwsze/złożone).

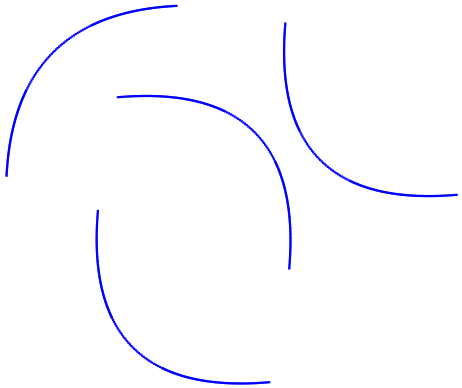
A jednak, rozpoznawanie problemów NP-zupełnych przez algorytmy probabilistyczne jest przypuszczalnie równie trudne, jak przez algorytmy deterministyczne.

Hipoteza derandomizacji Każdy probabilistyczny algorytm *wielomianowy* można symulować wielomianowym deterministycznym.

Czy można *wygenerować* obiekt losowy ?

struktura $\longmapsto$ chaos





$000 \longrightarrow 100101101$

$001 \longrightarrow 010010110$

$010 \longrightarrow 110001001$

$011 \longrightarrow 100001100$

$100 \longrightarrow 011010101$

$101 \longrightarrow 101100011$

$110 \longrightarrow 011100110$

$111 \longrightarrow 001000101$

kingsajz

Cel: wygenerować ciągi tak, by żaden algorytm wielomianowy (nawet probabilistyczny) nie potrafił ich odróżnić od ciągów losowych tej samej długości.

Algorytm deterministyczny

wejście $\longrightarrow$

0000	0100	1000	1100
0001	0101	1001	1101
0010	0110	1010	1110
0011	0111	1011	1111

$\longrightarrow$ tak/nie

Algorytm probabilistyczny

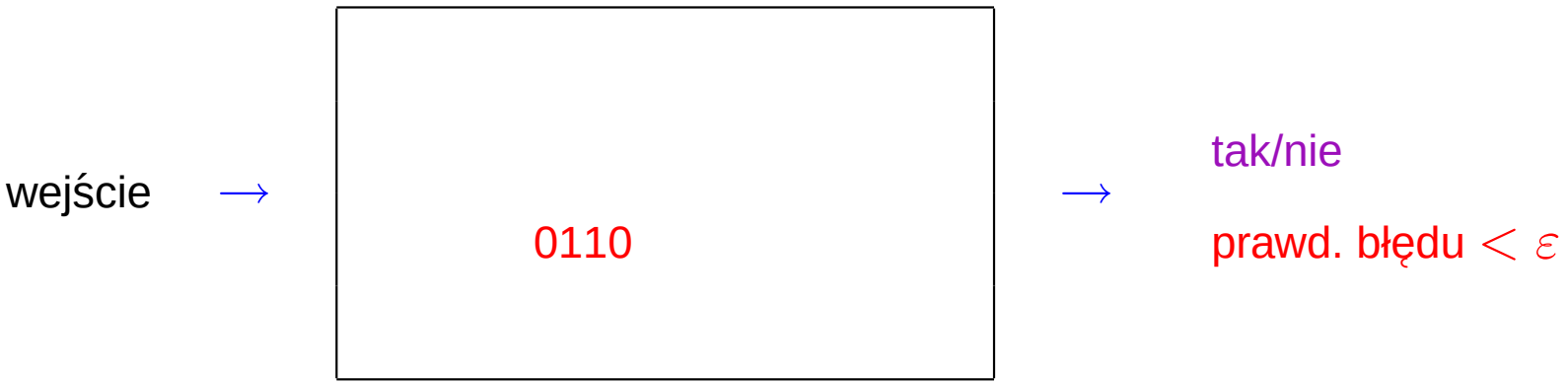
wejście $\longrightarrow$

0110

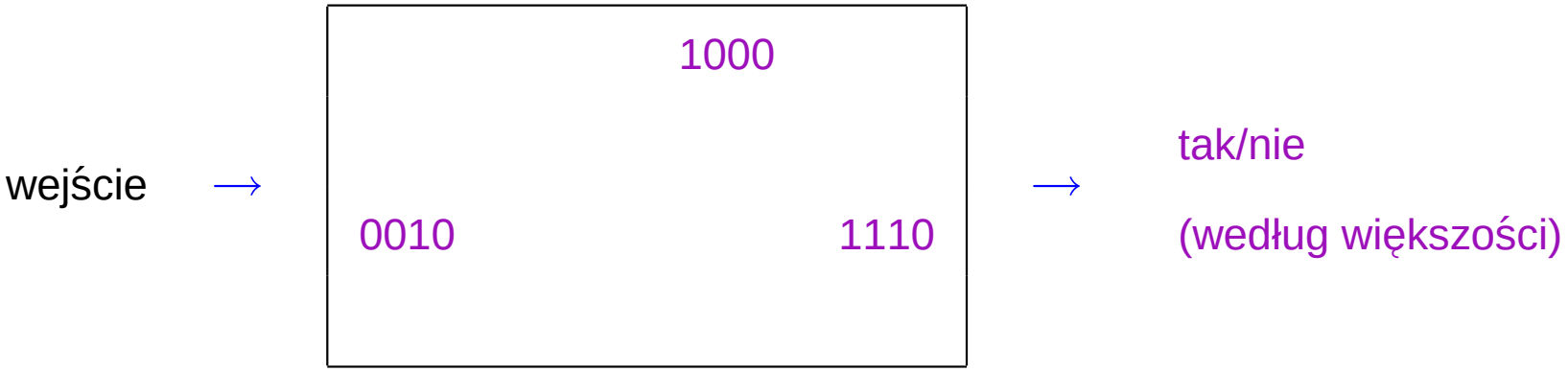
$\longrightarrow$ tak/nie

prawd. błędu $< \varepsilon$

Algorytm probabilistyczny



Algorytm probabilistyczny po derandomizacji (deterministyczny)



Czy jednak można wygenerować (prawie) losowy ciąg bitów ?

W sensie Kołmogorowa, ciąg 100101101011 jest losowy, jeśli nie da się wygenerować programem *krótszym* niż on sam.

(Zawsze można: *write* (100101101011).)

Np. ciąg

$$3.14159265358979323846264338327950288419716939937510$$
$$58209749445923078164062862089986280348253421170679$$
$$82148086513282306647093844609550582231725359408128$$

nie jest losowy.

Z definicji, funkcja

$$11010 \mapsto \underbrace{0110001111010110}_{losowy}$$

nie może być obliczalna.

Jeśli szyfr RSA jest bezpieczny, to last(w) jest kompletnie na podstawie RSA(w).

Czyli funkcja $w \mapsto \text{RSA}(w)\,\text{last}(w)$ jest prawie losowa.

Wtedy, startując z w_0, generujemy ciąg prawie losowy $a_1\, a_2\, a_3\, \ldots$, gdzie

$$
\begin{array}{llll}
w_0 & & & \\
a_1 & w_1 & & \\
 & a_2 & w_2 & \\
 & & a_3 & w_3 \\
 & & \ldots & \ldots
\end{array}
$$

$$a_{k+1} w_{k+1} = RSA(w_k)\text{last}(w_k)$$

A więc silne hipotezy kryptograficzne rugują losowość z obliczeń (teoretycznie).

Być może problemy NP-zupełne są problemami o naturze bliższej losowości, ,,bez teorii”.

SAT ???

Być może po prostu informacja potrzebuje czasu. . .

Być może problemy NP-zupełne są problemami o naturze bliższej losowości, ,,bez teorii”.

SAT ???

Być może po prostu informacja potrzebuje czasu. . .

It is never right to play ragtime fast.

Scott Joplin (1868–1917)