1) Quiz 

2) HW

Test 1: 2006 → web page.

ar 26

Berry's paradox
Komolgrov complexity
Godel's theory

$B(n)$ = no. of words to express n.

fixed dictionary of English

$B(2)$ = "two" = 1
$B(52)$ = "fifty two" = 2
$B(1048676)$ = 3 "two power twenty"

$S = \{ n \mid B(n) < 20 \}$ → finite = # words in dictionary.

$\bar{S} \in N$ → infinite.

b = min of $\bar{S}$: min # can't be expressed in < 20 words.

What is flawed? English is not formal lang.

⟨ one word → many meanings
⟨ express 1 thing → many ways

† Pick Java instead of English :

$K(n)$ = min number of bytes in Java to output n

$S = \{ n \mid KB(n) < 5G \}$

limit = 10G ;
i = i ; while $K(i) \le$ limit
i++
output (i)

$K(i)$

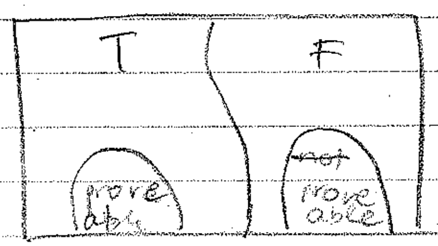
Upper: $\forall n$ $K(n) \le \lg n + c$
Lower: $K(n) \geq \lg n - c$ for most n.

Golder's Theorem :

→ In every consistent formal system that's strong enough,
There is a statement that is True but not provable.

Axiom : statement assumed → true.
Inference rules.

Consistent : possible to prove smth true / false



Hofstadtor : Gödel Eschor Bach
Math Drawing Music
- Self-reference

Chaitin
" $K(n) \geq \log n - c$ "

Program P enumerates proofs in F.

P(F) → loops all proofs of F in order
until find a proof :
$K(n) \geq \log n - c$ for $\log n > |F|$
output n.

$lgn - c \geq |F| + |P| \geq K(n) \geq lgn - c$

April 2 '09 Kolmogorov complexity of the word X....

~~min len~~ { p, p generates X }

+ 111111...11 → for (i=0; i < 1e6 ; i++)
print('1');

+ 01...0101... → for (i=0; i<5e5; i++)
print ('01');

+ print('010111...01001...')

- non-random sequences. $K(x) \leq len(x)$.
- random $K(x) \sim len(x)$.

+ Berry's paradox

smallest integer that can not be described in < 100 words.

→ Exactly this integer is described in ≤ 10 words.

→ Paradox.

computed by a program.

Proof:

```
x = 0;
found = true;

while (! found)
{
    if (K(x) > 100)
        found = true;
    else
        x++;
}
```

We count the # of char in p.
$\rightarrow$ estimate $K(11\ldots1) \leq 29$.
upper-bound

<u>NP</u> : class of problems : non-deterministic problems
$\rightarrow$ can't be solved in polynomial time

<u>Informally</u> : You can guess an answer and check it in poly. time.

<u>Formally</u> : 1) a feasible predicate $C(x, y)$
2) a polynomial

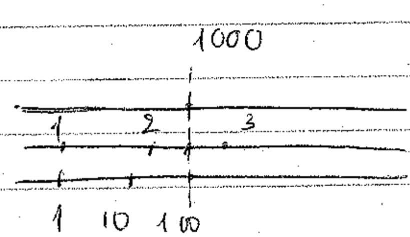
<u>Instances</u> : - x : given
- find y such that $C(x, y)$.
and $len(y) \leq P_e(len(x))$.

<u>Reduction</u> :

$$\begin{array}{ccc} X & \longrightarrow & y \\ \downarrow & & \\ X' & & y' \end{array}$$

$X = (X_1, X_2)$
$C(x, y) \equiv y = X_1 X_2$

Slide rule :



$X \notin (X_1, X_2)$

$c(X, Y) \equiv y = X_1 , X_2$

$X' = (X_1', X_2')$

$c(X', Y') = y' = X_1' + X_2'$

$$\begin{tikzcd}
X_1 \, X_2 \arrow[rr, bend left, "?"] \arrow[d, no head] & & X_1 + X_2 \\
\ln X_1 , \ell X_2 \arrow[rr] & & \ln X_1 + \ln X_2 \arrow[u]
\end{tikzcd}$$

- <u>3 algorithms :</u>

$$\begin{tikzcd}
X \arrow[r, bend left, "?"] \arrow[d, "U_1"'] & y \arrow[d, shift right, "u_2"'] \\
X' \arrow[r] & y' \arrow[u, shift right, "U_3"']
\end{tikzcd}$$

$U_1 (X_1, X_2) = (\log X_1 , \log X_2)$

if y' is the solution to the 2nd proplem ~~f~~
then $U_3'(y')$ solves the 1st problem.

If $c(u_1(X, y')$ is true,
then $c(X, U_3(y'))$ is also ~~blue~~ true.

$u_3(y') = 10^{y'}$.

$$\begin{tikzcd}
X_1 \arrow[d] & X_2 \arrow[d] & & X_1 \cdot X_2 \\
\log X_1 \arrow[dr] & \log X_2 \arrow[d] & & \\
& \log X_1 + \log X_2 \arrow[rr] & & \log X_1 \cdot X_2 \arrow[uu, "10^x"']
\end{tikzcd}$$

if $c(x, y)$ then $c'(u_1(x), u_2(y))$.

Ex: $at^4 + bt^2 + c = 0 \longrightarrow az^2 + bz^2 + c = 0$

$z = t^2$

$\begin{cases} X = (a, b, c) & : \text{given} \\ y = t & : \text{solution} \end{cases}$

$\Rightarrow \begin{cases} X' = (a, b, c) \\ y' = t^2 \ (z) \end{cases}$

$\begin{cases} u_2 = \text{square} \\ u_3 = \text{square root} \end{cases}$

+ A problem is NP hard if every problem from class NP can be reduced to it.

SAT: propositional satisfiability

Given: a Boolean formula $X_1 \ldots X_n$.

Quest: find a combination of values that make the formula True.

$(X_1 \vee \bar{X_2} \vee X_3) \wedge (\bar{X_1} \vee X_2 \vee X_3)$

$\rightarrow \begin{cases} X_1 = T \\ X_2 = T \\ X_3 = F \end{cases}$

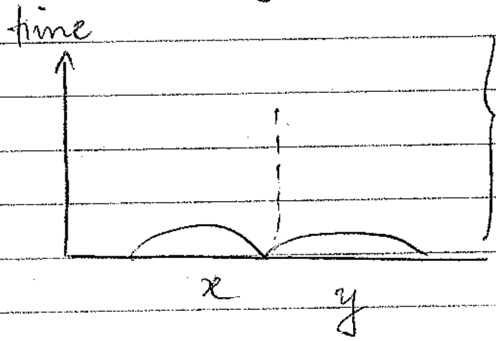
We have an ~~set~~ instance of a problem NP

Given: x

$c(x, y)$ poly. time.

We want to find y

$C(x, y)$ is true.
$len(y) \leq P_e(len(x))$.



$\leq$ poly. of $len(x)$ and $len(y)$

S - largest # of possible states

Δt $\quad$ $S_{i,t}$ - state of i-th cell at t-th moment.

$S_{i,b,t}$ - b-th bit in i-th cell at t-th moment

April 7th $\quad C(x, y)$

Given : x

Find : $y > t$

$C(x, y) \wedge len(y) \leq P_e(len(x))$

Comp. device

input : x, y

Output : T or F.

Δt - time quantum

ΔV - smallest size of the cell.

S - largest states in a cell.

S_{it} - state of cell i at moment t.

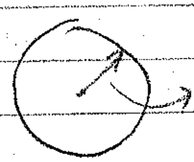
B − # of bits to describe an individual state.

$$2^B \geq S \simeq 2^{\log_2 S}$$

$$B \geq \log_2 S$$
$$B = \lceil \log_2 S \rceil$$

S small − $s_{i,b,t}$ be b-th bit in the binary descr. of the state $S_{i,t}$.

$S_{i,t+1}$: depends on the state of different cells in previous moment of time.



$$r = \underset{\substack{\| \\ \text{light speed}}}{c} \times \Delta t$$

$$N_{neighbor} \leq \frac{V}{\Delta V} = \frac{4/3\, \pi c^3 \cdot \Delta t^3}{\Delta V}$$

→ speed of light.

$$S_{i,t+1} = f_{i,t}\big(\underbrace{S_{i,t} \cdots S_{j,t}}_{\leq N_{neigh}}\big)$$

$$s_{i,b,\underbrace{t+1}_{t+1}} = f_{i,b,\underbrace{t}_{t}}\big(s_{i,t} \cdots s_{j,b,t}\big)$$

→ depends on the previous state of cell

(+) cell: any thing . . transistor ....

$S_{i,2,t}$ $S_{i,1,t}$

adder

$$\begin{array}{r} 101001 = S_{i,t} \\ 010111 = S_{j,t} \\ \hline \text{-----}\ 0 \rightarrow S_{i,t+1} \end{array}$$

(1) CNF form

$\left.\begin{array}{l} X_1 \dots X_n \\ \bar{X}_1 \dots \bar{X}_n \end{array}\right\}$ literals

$(a \vee b \vee c)$ $(a \vee b)$

$(X_1 \vee \bar{X}_2 \vee X_3) \wedge (\bar{X}_1 \vee X_2) \wedge \dots$

$f(x_1, x_2) \equiv (X_1 = X_2)$

X_1	X_2	f
F	F	T
F	T	F
T	F	F
T	T	T

$f \equiv (\bar{X}_1 \wedge \bar{X}_2) \vee (X_1 \wedge X_2)$

At the end,

$F_{1,1,1}$ & $F_{1,2,1}$ & ... & $F_{N,B,T}$

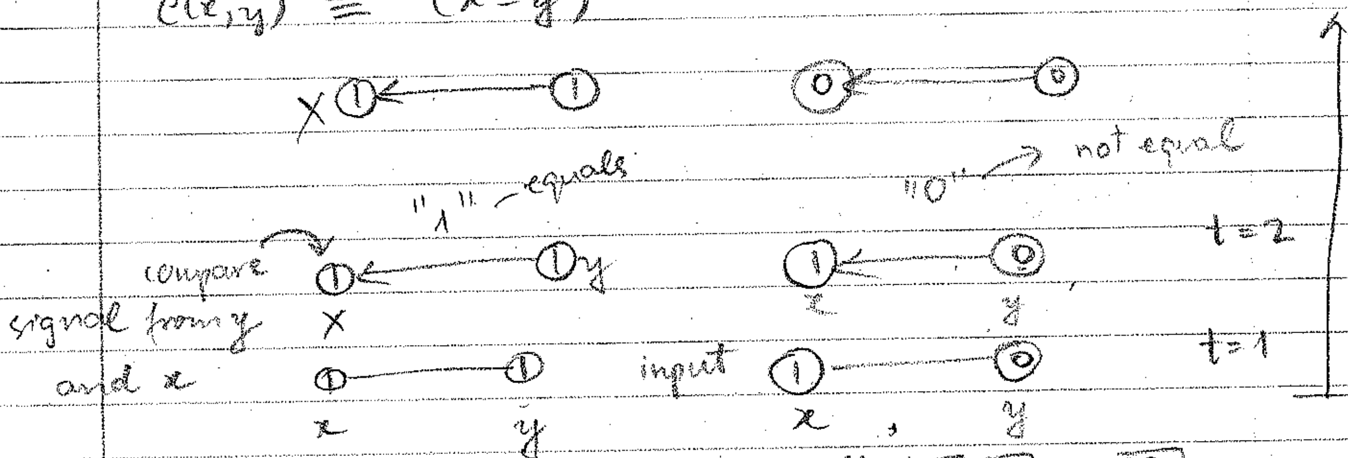
Time $T \leq P_c(\text{len}\,x + \text{len}\,y) \leq P_c(n)$

$$N \leq \frac{V}{\Delta V} = \frac{\frac{4}{3}\pi c^3 \cdot T^3}{\Delta V}$$

$(S_{s_1, b_1, 1} = X_1)$ & $(\ldots\ldots)$ ... & $S_{s_i, b_i, T} = 1$

NP $\longrightarrow$ computational device to check

Example :
x - 1 bit
y - 1 bit
$C(x,y) \equiv (x = y)$ ← equals



→ cell $\boxed{1}$ $\boxed{2}$ $\boxed{3}$

3 cells : any part = x, y, 1 wire

State : 1, 0, No signal → $\boxed{S = 3}$

2 bits to represent the states

cell X : 00 01
cell y : 00 01

wire : no signal 00 } S_0
signal "0" 10 } S_1
signal "1" 11 } S_2

ibt

$S_{112} = 0$	$S_{212} = 0$
$S_{122} = S_{121}$	$S_{222} = S_{221}$

$S_{312} = 1$

$S_{322} = S_{221}$

$\neg S_{112} \;\&\; (\bar{S}_{122} \vee S_{222}) \;\&\; (S_{122} \vee \bar{S}_{222}) \;\&$

$(\neg S_{212} \;\&\; S_{321} \;\&\; (\bar{S}_{322} \vee S_{221}) \;\&\; (S_{322} \vee \bar{S}_{221})$

$\&\; (\neg S_{222} \vee S_{221}) \;\&\; (\neg S_{221} \vee S_{222}) \;\ldots\ldots$

$S_{213} = 0$

$S_{123} = \begin{cases} 1 & \text{if } S_{322} = S_{122} \\ 0 & \text{otherwise} \end{cases}$

$S_{223} = S_{222}$

transition

$S_{313} = 0$

$f \longrightarrow \boxed{a = \begin{cases} 1 & \text{if } b = c \\ 0 & \text{else} \end{cases}}$

$S_{323} = 0$

$S_{113} = 0$

a	b	c	f	$\bar{f}$
0	0	0	0	1
0	0	1	1	0
0	1	0	1	0
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	0
0	1	1	0	1

$\bar{f} = (\bar{a} \,\&\, \bar{b}\,\bar{c}) \vee (a \,\&\, \bar{b} \,\&\, c) \vee$
$(a \,\&\, b \,\&\, \bar{c}) \vee (\bar{a} \,\&\, b \,\&\, c)$

$$f = (a \vee b \vee c) \,\&\, (a \vee \bar{b} \vee \bar{c}) \,\&\, (\bar{a} \vee b \vee \bar{c}) \,\&\, (\bar{a} \vee \bar{b} \vee c).$$

 : Extra credit

check if $X \rightarrow Y \Leftrightarrow Y \vee \neg X$