

CS 121: Lecture 20

Cook-Levin Theorem

Madhu Sudan

<https://madhu.seas.harvard.edu/courses/Fall2020>

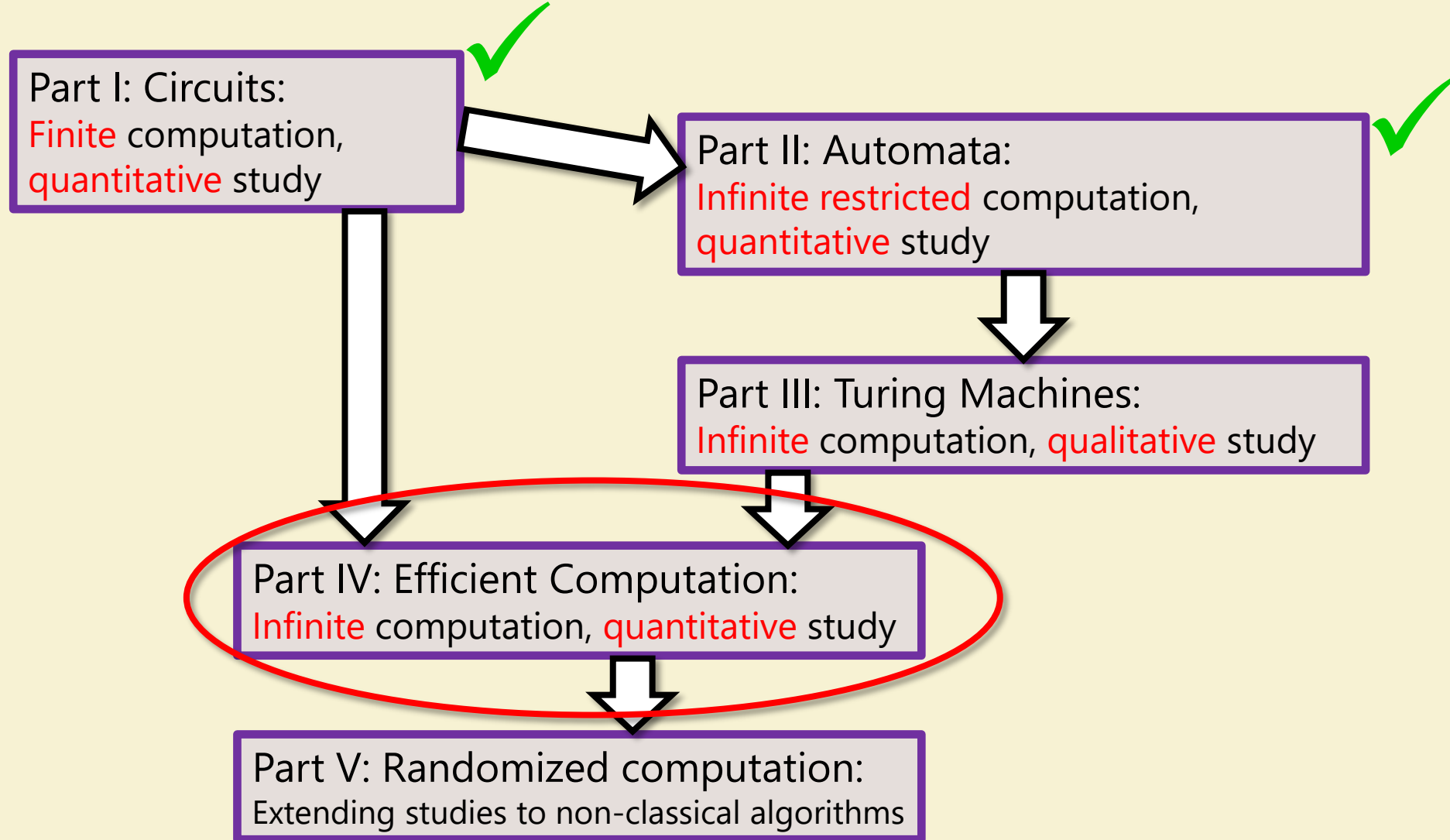
Book: <https://introtcs.org>

How to contact us { The whole staff (faster response): [CS 121 Piazza](#)
Only the course heads (slower): cs121.fall2020.course.heads@gmail.com

Announcements:

- Advanced section Thursday: Nicole Immorlica on EconCS
- Midterm 2 in 1 week.
 - Open book (Barak only). 2 pages cheat sheet (no collaboration).
 - 90 minutes long. (70 if handwritten.)
- Homework 5 due Thursday.

Where we are:



Review of last lecture

- Defined NP (solutions/witnesses/proofs polytime verifiable)
 - $F: \{0,1\}^* \rightarrow \{0,1\} \in \text{NP} \Leftrightarrow \exists V_F \text{ polytime comp. s.t. } F(x) = 1 \Leftrightarrow \exists w \text{ s.t. } V_F(x, w) = 1$
- Defined NP-Hard and NP-complete
 - $F \text{ NP-hard} \Leftrightarrow \forall G \in \text{NP}, G \leq_p F$
 - $F \text{ NP-complete} \Leftrightarrow F \in \text{NP} \text{ and } F \text{ NP-complete.}$
- Asserted: 3SAT is NP-Complete
 - $3\text{SAT}(C_1 \wedge C_2 \cdots C_m) = 1 \Leftrightarrow \exists x_1, \dots, x_n \in \{0,1\} \text{ s.t. } \forall j \exists \text{ literal in } C_j \text{ that is } 1.$
 - Will prove today
- Proved: ISET is NP-Complete (assuming assertion)

Today

- Cook-Levin Theorem: 3SAT is NP-Complete

NP-Completeness historically

- Many mathematicians sensed NP-hardness:
 - Gauss (1800s): Can you factor integers?
 - Godel (1956): Can you automate proving of theorems?
 - Edmonds (1967): Travelling Salesperson has no polynomial time algorithm?

*If [3SAT has $O(n^2)$ time algorithm] then in spite of the undecidability of the Entscheidungsproblem, the mental work of a mathematician concerning Yes-or-No questions could be **completely replaced by a machine***

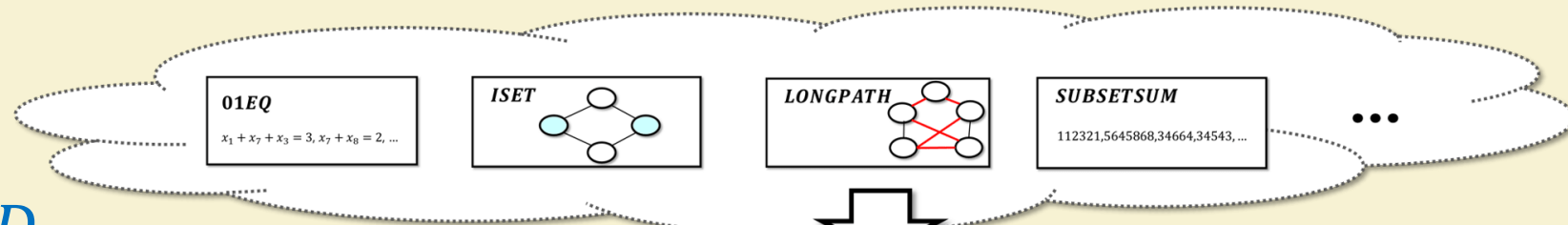
Kurt Gödel to John von-Neumann, 1956

Today's Theorem

$3SAT$ is NP -hard.
 NP -hard + in $NP = NP$ complete

Cook Levin Theorem: $\forall F \in NP \ F \leq_p \ 3SAT$

Proof:

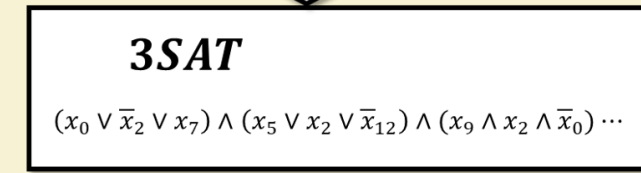
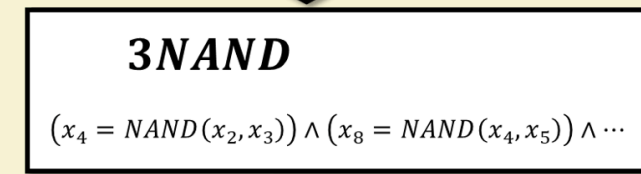
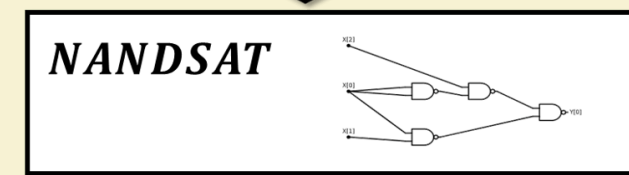


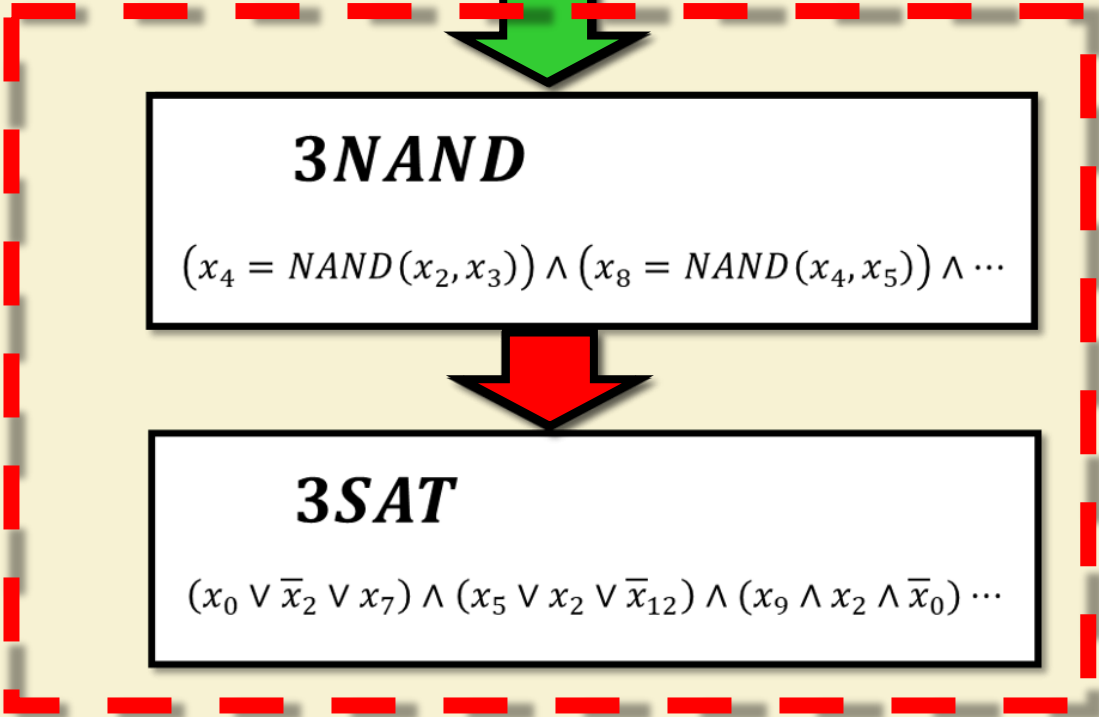
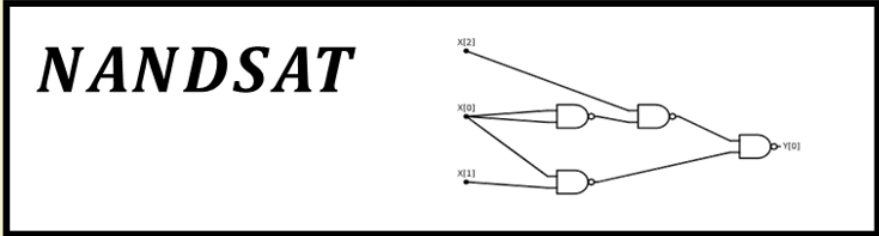
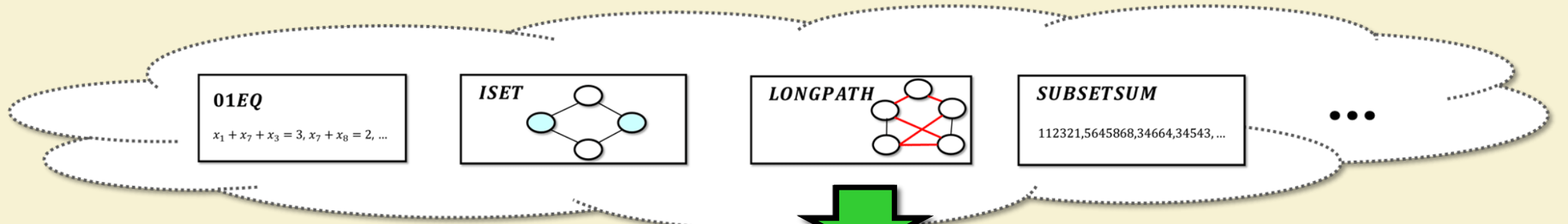
Define $NANDSAT, 3NAND$

Lemma 1: $\forall F \in NP, F \leq_p \ NANDSAT$

Lemma 2: $NANDSAT \leq_p \ 3NAND$

Lemma 3: $3NAND \leq_p \ 3SAT$





3NAND

Input: Ψ is AND of constraints of form $z_i = \text{NAND}(z_j, z_k)$

Output: 1 iff there is assignment $z \in \{0,1\}^r$ satisfying Ψ

Q: If $\Psi = (z_0 = \text{NAND}(z_2, z_3)) \wedge (z_3 = \text{NAND}(z_2, z_1)) \wedge (z_1 = \text{NAND}(z_2, z_3))$
what is $3\text{NAND}(\Psi)$?

Lemma 2: $\text{NANDSAT} \leq_p 3\text{NAND}$

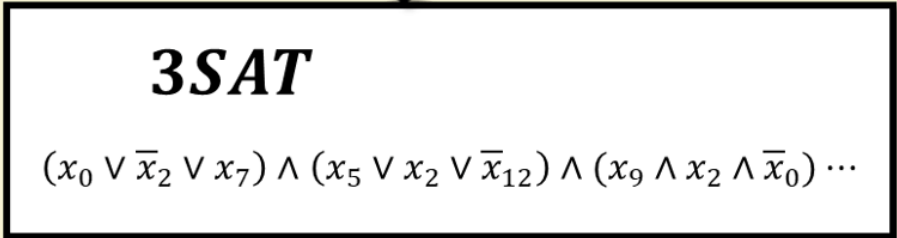
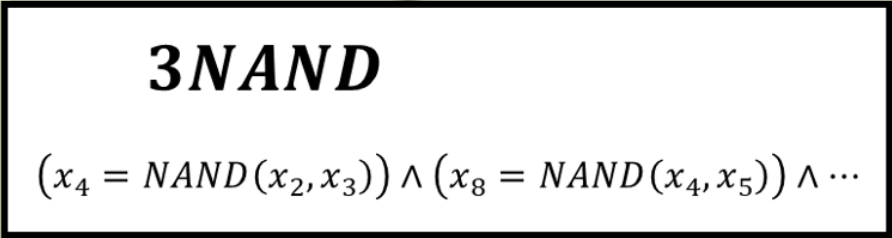
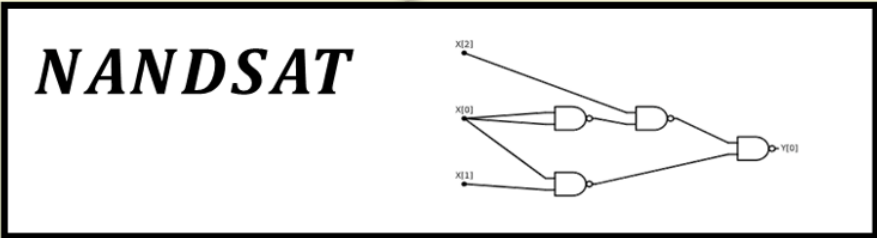
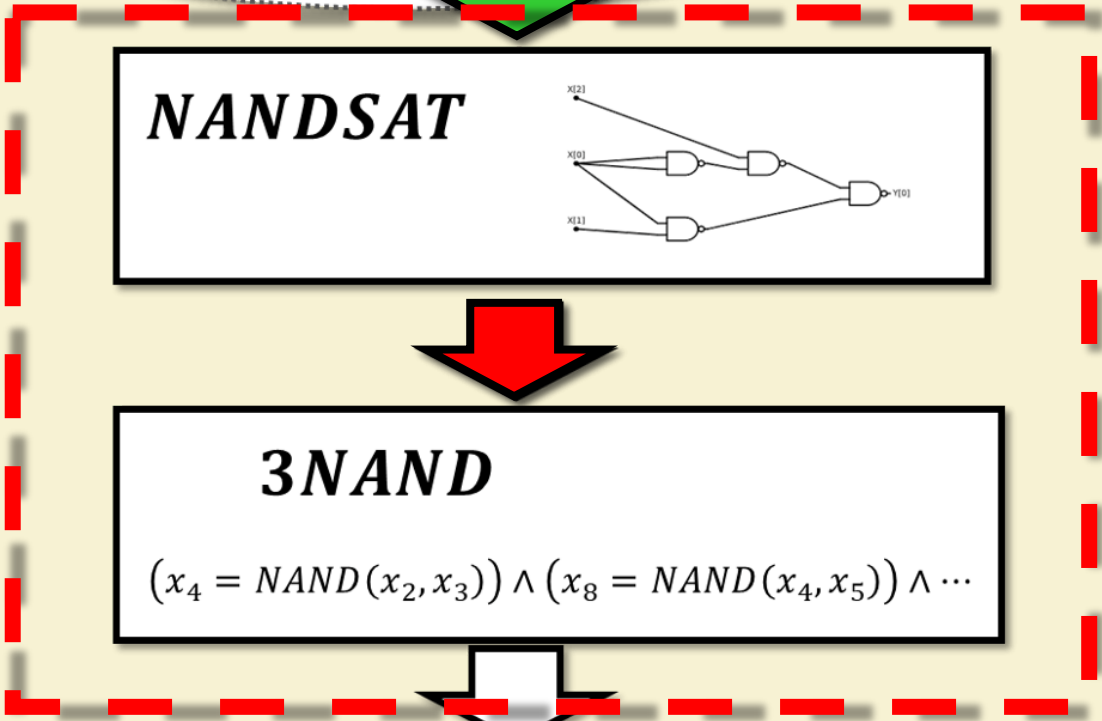
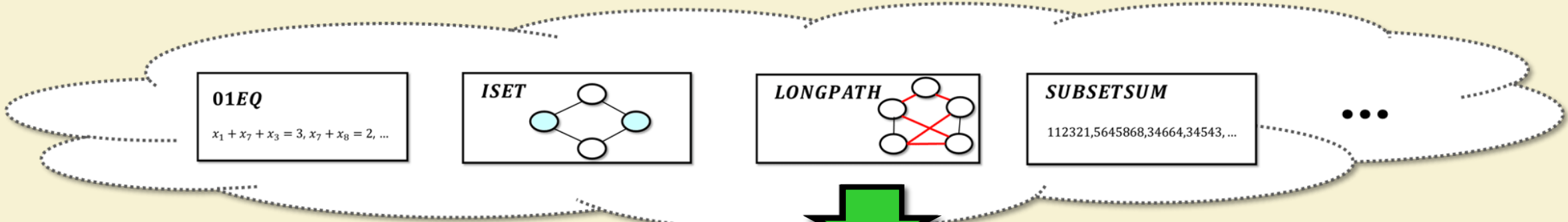
Lemma 3: $3NAND \leq_p 3SAT$

Key claim: For every $a, b, c \in \{0,1\}$,

$$c = NAND(a, b) \iff \begin{matrix} \bar{a} \vee \bar{b} \vee \bar{c} \\ \wedge \\ c \vee a \\ \wedge \\ c \vee b \end{matrix}$$

Q: Prove key claim.

Key claim \Rightarrow lemma.

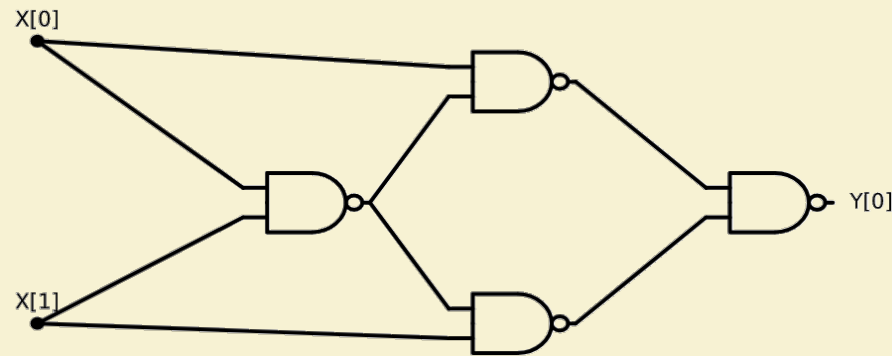


NANDSAT

Input: NAND-CIRC program P (aka circuit with NAND gates)

Output: 1 iff there is $x \in \{0,1\}^n$ s.t. $P(x) = 1$

Q: What is $NANDSAT($

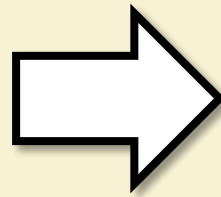
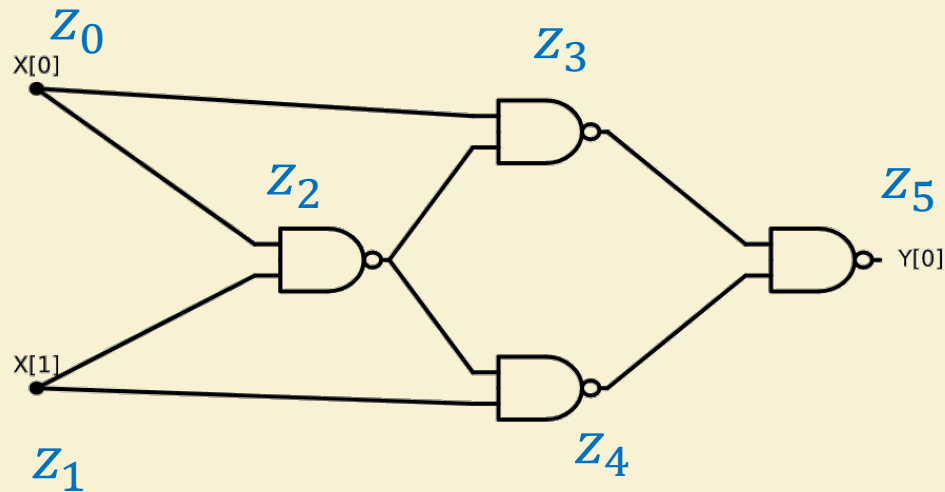


Input: Ψ is AND of constraints of form $z_i = \text{NAND}(z_j, z_k)$

Output: 1 iff there is assignment $z \in \{0,1\}^r$ satisfying Ψ

Lemma 2: $\text{NANDSAT} \leq_p 3\text{NAND}$

“Proof by example:”



$$z_2 = \text{NAND}(z_0, z_1)$$

$$z_3 = \text{NAND}(z_0, z_2)$$

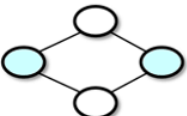
$$z_4 = \text{NAND}(z_1, z_2)$$

$$z_5 = \text{NAND}(z_3, z_4)$$

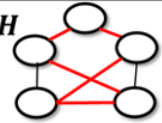
$$z_5 = 1 \quad \left\{ \begin{array}{l} z_6 = \text{NAND}(z_0, z_0) \\ z_5 = \text{NAND}(z_0, z_6) \end{array} \right.$$

01EQ
 $x_1 + x_7 + x_3 = 3, x_7 + x_8 = 2, \dots$

ISET



LONGPATH

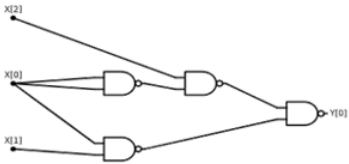


SUBSETSUM
112321,5645868,34664,34543, ...

...



NANDSAT



3NAND

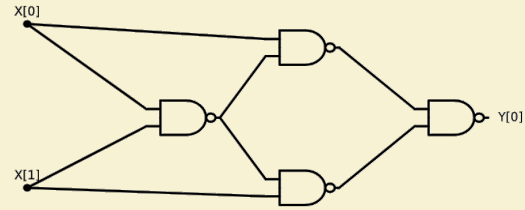
$(x_4 = \text{NAND}(x_2, x_3)) \wedge (x_8 = \text{NAND}(x_4, x_5)) \wedge \dots$



3SAT

$(x_0 \vee \bar{x}_2 \vee x_7) \wedge (x_5 \vee x_2 \vee \bar{x}_{12}) \wedge (x_9 \wedge x_2 \wedge \bar{x}_0) \dots$

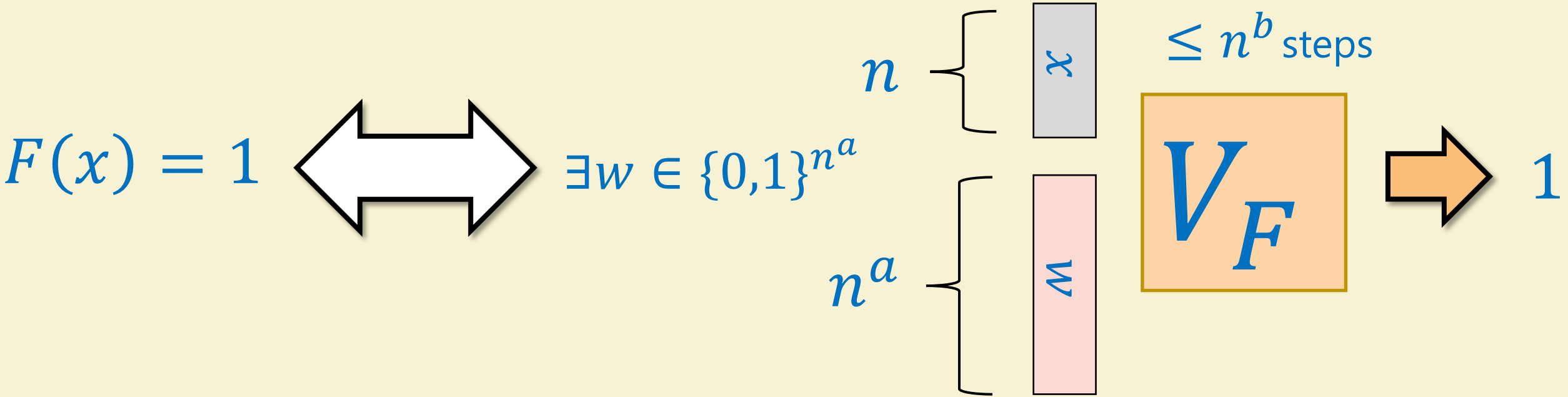
Input: NAND-CIRC program P (aka circuit with NAND gates)



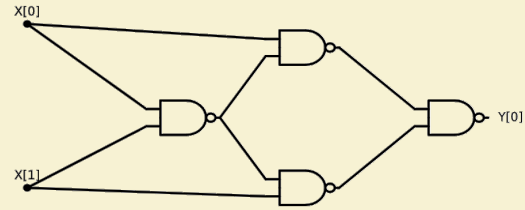
Output: 1 iff there is $x \in \{0,1\}^n$ s.t. $P(x) = 1$

Lemma 1: For every $F \in NP$, $F \leq_p NANDSAT$

Proof: If $F \in NP$ we know \exists poly-time TM V_F s.t. $\forall x \in \{0,1\}^n$



Input: NAND-CIRC program P (aka circuit with NAND gates)

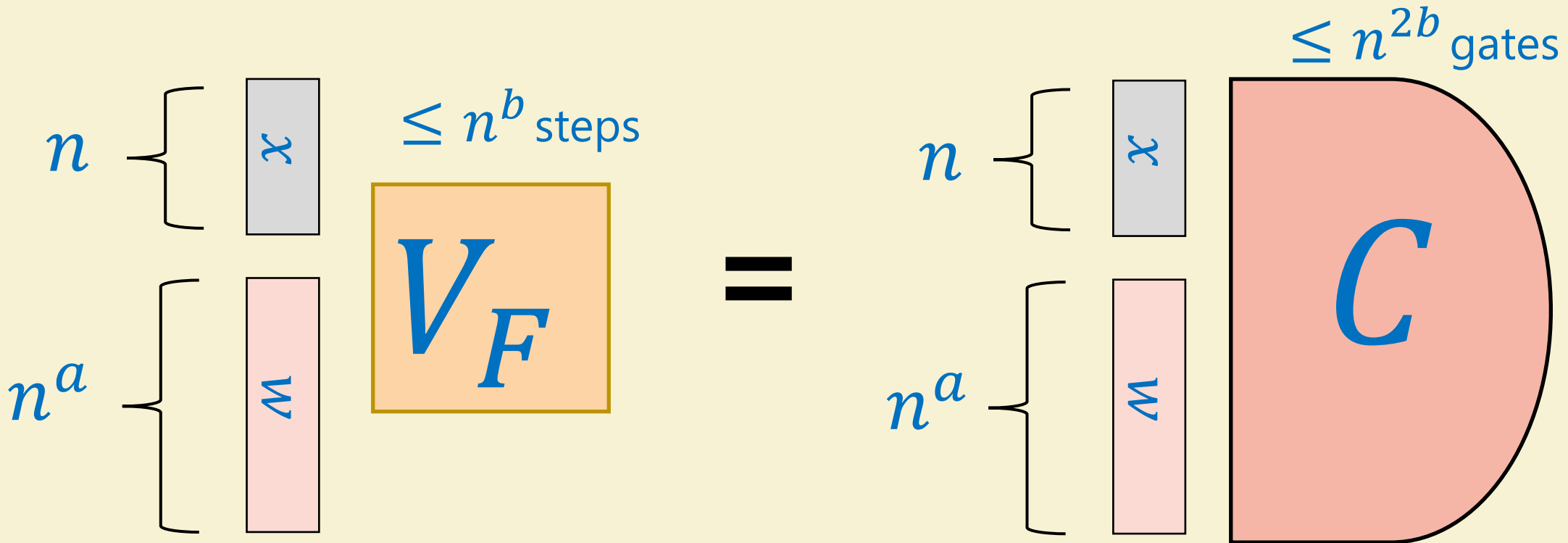


Output: 1 iff there is $x \in \{0,1\}^n$ s.t. $P(x) = 1$

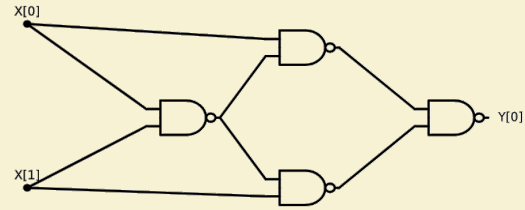
Lemma 1: For every $F \in NP$, $F \leq_p NANDSAT$

Proof: If $F \in NP$ we know \exists poly-time TM M s.t. $\forall x \in \{0,1\}^n$

By proof of $P \subseteq P_{/poly}$ (Time $\leq \approx$ SIZE) can find n^{2b} sized circuit C s.t.



Input: NAND-CIRC program P (aka circuit with NAND gates)



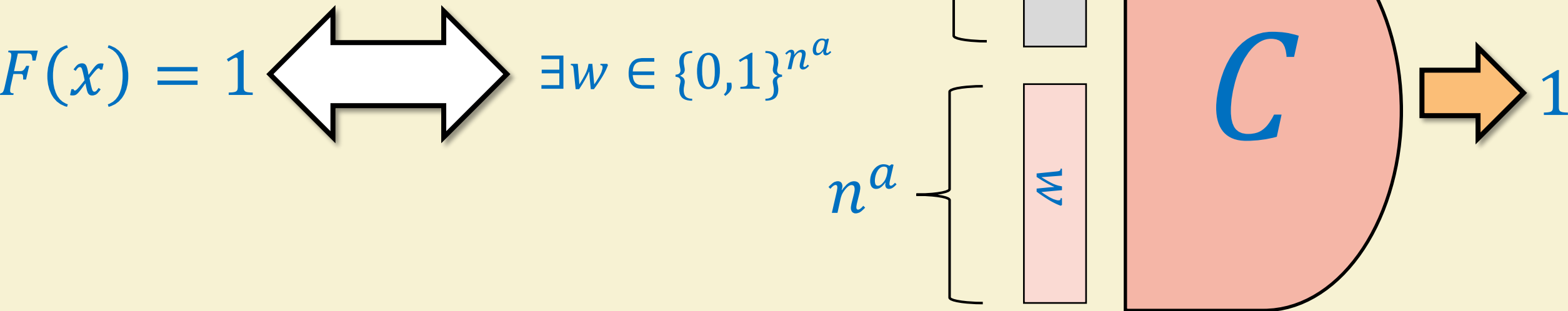
Output: 1 iff there is $x \in \{0,1\}^n$ s.t. $P(x) = 1$

Lemma 1: For every $F \in NP$, $F \leq_p NANDSAT$

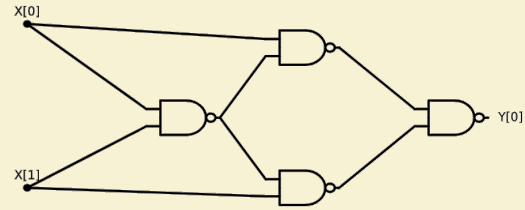
Proof: If $F \in NP$ we know \exists poly-time TM V_F s.t. $\forall x \in \{0,1\}^n$

By proof of $P \subseteq P_{/poly}$ can find n^{2b} sized circuit C s.t.

$\leq n^{2b}$ gates



Input: NAND-CIRC program P (aka circuit with NAND gates)



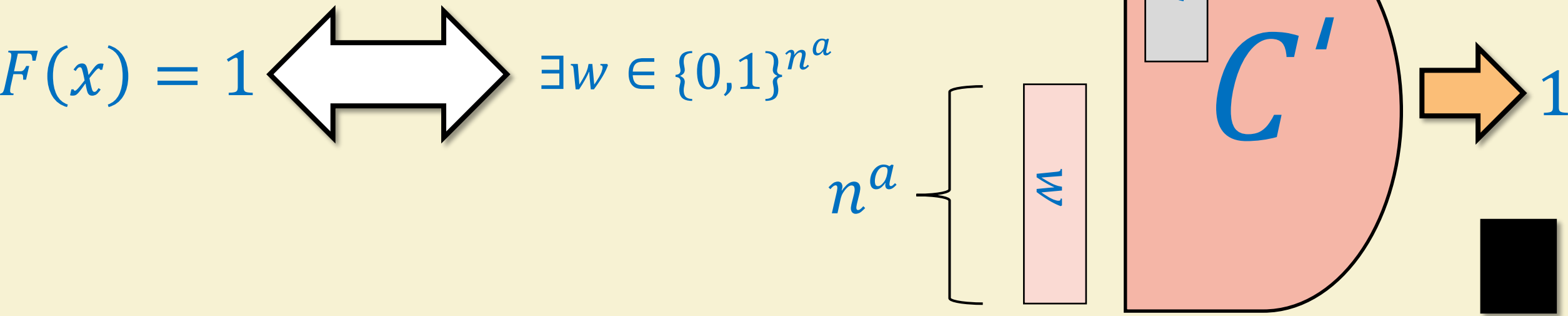
Output: 1 iff there is $x \in \{0,1\}^n$ s.t. $P(x) = 1$

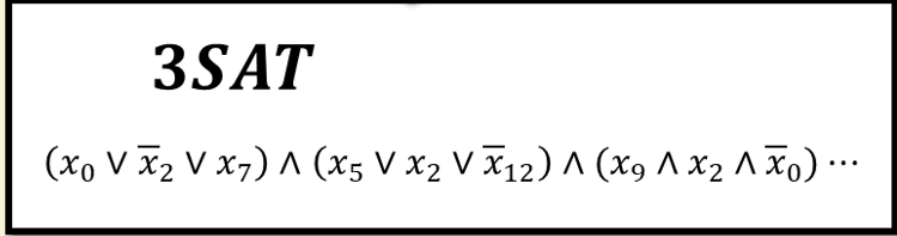
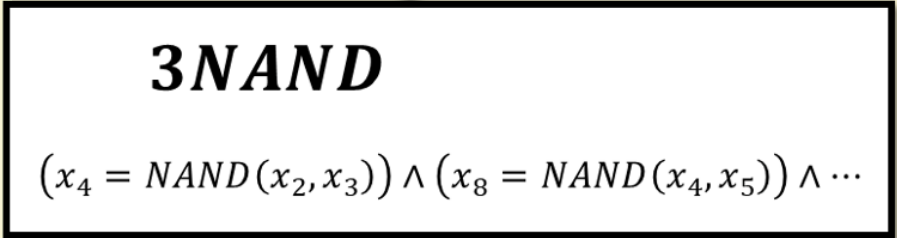
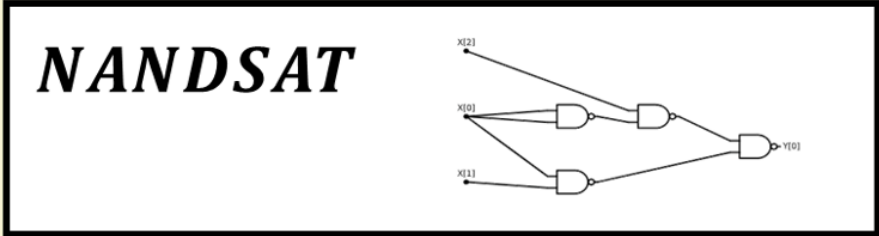
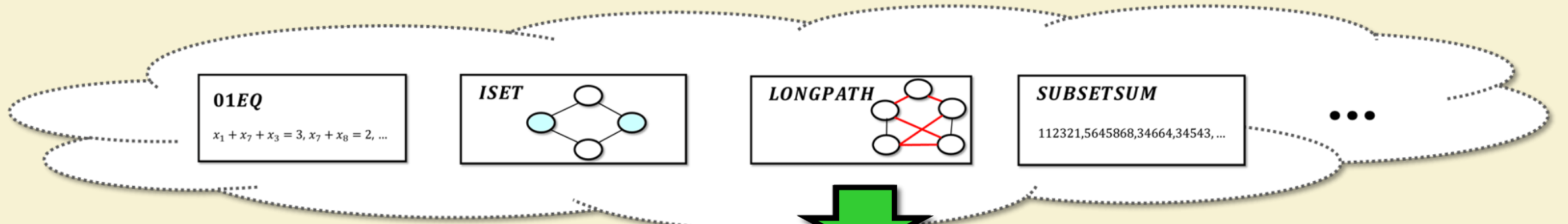
Lemma 1: For every $F \in NP$, $F \leq_p NANDSAT$

Proof: If $F \in NP$ we know \exists poly-time TM M s.t. $\forall x \in \{0,1\}^n$

By proof of $P \subseteq P_{/poly}$ can find n^{2b} sized circuit C s.t.

$\leq n^{2b}$ gates





Example to illustrate proof

Example: $ISET \leq_p 3SAT$

Input: Graph G integer k

Step 1: NAND circuit C_G such that

$C_G(S) = 1$ iff $|S| \geq k$ and S independent in G

Step 2: 3NAND formula Ψ s.t. $\exists z$ with $\Psi(z) = 1$ iff $\exists S$ s.t. $C_G(S) = 1$

Step 3: 3CNF formula φ s.t. $\varphi(z) = \Psi(z)$ for every z

Next lecture

- More reductions. See more diverse NP-complete problems.