Lesson 2

Computability

Math 574 - Topics in Logic
Penn State, Spring 2014

Jan Reimann

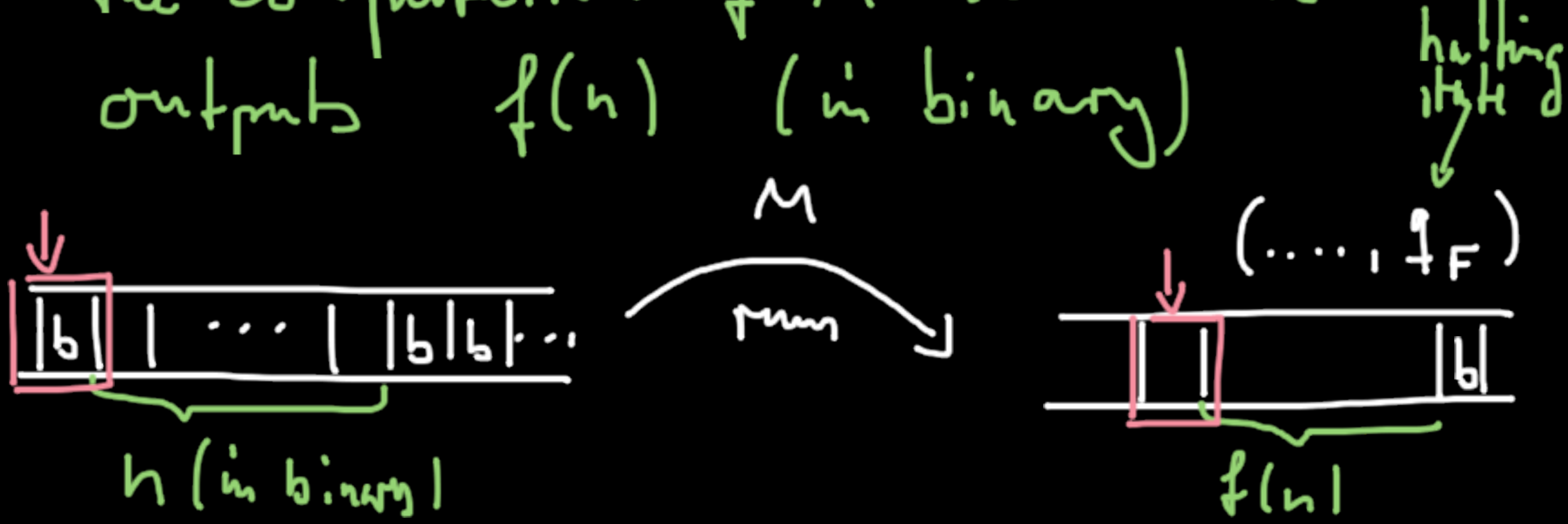
PENNSTATE
Eberly College of Science

2-4

Computable Functions

Turing computable functions

DEF: $f: \mathbb{N} \to \mathbb{N}$ is Turing computable if there exists a TM M s.t. on input n (given as binary representation), the computation of M halts and outputs $f(n)$ (in binary)



Decidable problems

DEF: A set $X \subseteq \mathbb{N}$ is Turing decidable if its characteristic function

$$\chi_X(n) = \begin{cases} 1 & \text{if } n \in X \\ 0 & \text{if } n \notin X \end{cases}$$

is Turing computable.

- These concepts can also be defined for $f: A^{<\mathbb{N}} \to A^{<\mathbb{N}}$, or $X \subseteq A^{<\mathbb{N}}$

EXAMPLES

T-computable

- $f(n) = 2n$
- $f(n) =$ n-th digit in binary expansion of π

T-decidable

- $L = \{0^n 1^n : n \geq 0\}$
- $P = \{p : p \text{ prime}\}$

The CHURCH - TURING Thesis

Notions of computability equivalent to T-computability

- λ-calculus (Church)
- μ-recursive functions (Gödel - Herbrand)
- register machines (various authors)

<u>C-T Thesis</u>: A function $f : \mathbb{N} \to \mathbb{N}$ is <u>algorithmically computable</u> iff it is <u>Turing computable</u>

Partial computable functions

- TM may not halt at all when running on an input.

$$(\square, *, *, R, \square)$$

DEF: $f : X \to \mathbb{N}$, $X \subseteq \mathbb{N}$, is partial comput. if there ex. TM M s.t.

M halts on input $n \iff n \in X$

and if $n \in X$, outputs $f(n)$

Semidecidable sets

recursively enumerable

DEF: $Z \subseteq \mathbb{N}$ is **semidecidable** if the function

$$\varphi_Z(n) = \begin{cases} 1 & \text{if } n \in Z \\ \uparrow & \text{if } n \notin Z \end{cases}$$

undefined

is partial computable.

i.e. $\exists$ TM M s.t.

M halts on n (and outputs 1) $\iff n \in Z$

EXAMPLE

$p_0, p_1, p_2, p_3, \dots$

enumeration of all polynomials

$p_j(x, y, z)$ with integer coefficients

$Z = \{ j : \; p_j = 0$ has solution in the integers $\}$