

Logic (master program)

Georg Moser

Institute of Computer Science @ UIBK

Winter 2008



Primitive Recursive Functions

Definition closed under primitive recursion

let \mathcal{S} be a set of functions on \mathbb{N} and suppose

- $\forall h: \mathbb{N}^{n-1} \rightarrow \mathbb{N}$ in \mathcal{S}
- $\forall g: \mathbb{N}^{n+1} \rightarrow \mathbb{N}$ in \mathcal{S}

$n > 0$

the function defined as:

$$f(0, x_2, \dots, x_n) = h(x_2, \dots, x_n)$$

$$f(x_1 + 1, \dots, x_n) = g(x_1, \dots, x_n, f(x_1, \dots, x_n))$$

is contained in \mathcal{S} , then \mathcal{S} is **closed under primitive recursion**

Definition

the **primitive recursive functions** are the smallest set containing the basic functions that is closed under composition and primitive recursion

Summary of Last Lecture

Definition

basic functions

the functions Z, s, p_i^n are called **basic functions**

Definition

closed under composition

let \mathcal{S} be a set of functions on \mathbb{N} and suppose

- $\forall h: \mathbb{N}^m \rightarrow \mathbb{N}$ in \mathcal{S}
- $\forall 1 \leq i \leq m \ g_i: \mathbb{N}^n \rightarrow \mathbb{N}$ in \mathcal{S}

the function defined as:

$$f(x_1, \dots, x_n) = h(g_1(x_1, \dots, x_n), \dots, g_m(x_1, \dots, x_n))$$

is contained in \mathcal{S} , then \mathcal{S} is **closed under composition**

Recursive Functions

Definition closed under unbounded search

let \mathcal{S} be a set of functions on \mathbb{N} and suppose

- $\forall f: \mathbb{N}^{n+1} \rightarrow \mathbb{N}$ in \mathcal{S}

the function defined as:

$$\mu_f(x_1, \dots, x_n) = \begin{cases} z & \forall y \leq z \ f(\vec{x}, y) \text{ is defined and} \\ & z = \min\{v \mid f(\vec{x}, v) = 0\} \\ \text{undefined} & \text{otherwise} \end{cases}$$

is contained in \mathcal{S} , then \mathcal{S} is **closed under unbounded search**

Definition

the set of **recursive functions** is the smallest set containing the primitive recursive functions that is closed under unbounded search

Computable Sets and Relations

Definition

characteristic function

the **characteristic function** χ_A of $A \subseteq \mathbb{N}^n$:

$$\chi_A(x_1, \dots, x_n) = \begin{cases} 1 & (x_1, \dots, x_n) \in A \\ 0 & (x_1, \dots, x_n) \notin A \end{cases}$$

Definition

set $A \subseteq \mathbb{N}^n$ is called

- **primitive recursive** if χ_A is primitive recursive
- **recursive** if χ_A is recursive

Proposition

if A is definable by a quantifier-free \mathcal{V}_{ar} -formula, then A is primitive recursive

Recursive Functions

let $\exists y(y < x \wedge \varphi(x, y))$ abbreviate $\exists y \exists z(y + z = x \wedge z \neq 0 \wedge \varphi(x, y))$

Definition

closed under bounded quantifiers

let \mathcal{F} be a set of \mathcal{V}_{ar} -formulas and suppose

- $\forall \varphi(x, y) \in \mathcal{F}$

also $\exists y(y < x \wedge \varphi(x, y))$ is contained in \mathcal{F} then \mathcal{F} is **closed under bounded quantifiers**

Definition

Δ_0 is the smallest set of \mathcal{V}_{ar} -formulas containing the atomic formulas that is closed under negation, conjunction and under bounded quantifiers

Proposition

A is definable by a Δ_0 -formula if and only if A is primitive recursive

Content

introduction, propositional logic, semantics, formal proofs, resolution (propositional)

first-order logic, semantics, structures, theories and models, formal proofs, Herbrand theory, completeness of first-order logic, properties of first-order logic, resolution (first-order)

introduction to computability, introduction to complexity, finite model theory

beyond first order: modal logics in a general setting, higher-order logics, introduction to Isabelle

Proof

we only show direction from left to right; let $\varphi(x, y)$ be a \mathcal{V}_{ar} -formula

- suppose $\varphi(x, y)$ defines a primitive recursive set A
- as $\chi_A(x, y)$ is primitive recursive:

$$\text{sum}_{\chi_A}(x, y) = \sum_{z < y} \chi_A(x, z)$$

is primitive recursive

- $g(x) := \text{sum}_{\chi_A}(x, x)$ is primitive recursive
- observe

$$1 \dot{-} g(x) = \begin{cases} 0 & \text{if } \exists z < x \chi_A(x, z) = 1 \\ 1 & \text{otherwise} \end{cases}$$

- if B is defined by $\exists y(y < x \wedge \varphi(x, y))$ then $\chi_B = 1 \dot{-} (1 \dot{-} g(x))$
- B is primitive recursive ■

direction from right to left is instrumental in the proof of Gödel's Incompleteness Theorem

Semi-decidable Decision Problems

Definition recursive enumerable

$A \subseteq \mathbb{N}^n$ is **recursively enumerable** if \exists total recursive $f: \mathbb{N} \rightarrow \mathbb{N}^n$ such that

$$A = \{f(0), f(1), f(2), \dots\}$$

Lemma

let $A \subseteq \mathbb{N}^n$ and let h_A be defined as

$$h_A(x_1, \dots, x_n) = \begin{cases} 1 & (x_1, \dots, x_n) \in A \\ \text{undefined} & \text{otherwise} \end{cases}$$

A is recursive enumerable if and only if h_A is partial recursive

Notation

let f and g be **partial** functions; we write $f(x_1, \dots, x_n) \simeq g(x_1, \dots, x_n)$ if the functions have the same domain and $f(x_1, \dots, x_n) = g(x_1, \dots, x_n)$ for any \vec{x} in this domain

Proof

we only show the direction left to right and assume $n = 1$

- \exists total recursive f , such that A is range of f
- define $g(x, y) = (f(x) \div y) + (y \div f(x))$
- then $g(x, y) = 0$ if and only if $f(x) = y$
- A is domain of $\mu_x g(x, y)$ which is (partial) recursive
- hence $h_A(x) \simeq c(\mu_x g(x, y))$, where $c(x) = 1$

decision problem A	set $A \subseteq \mathbb{N}^n$	function
decidable	recursive	χ_A is recursive
semi-decidable	r.e.	h_A is partial recursive
undecidable	not recursive	χ_A is not recursive

Example

the n -ary recursive functions are recursively enumerable:

$$\varphi_0^n, \varphi_1^n, \varphi_2^n, \varphi_3^n, \dots$$

Undecidable Decision Problems

Lemma

- $\forall n, m$
- \exists total primitive recursive function S_n^m such that

$$\varphi_{S_n^m(e, \vec{x})}^m(y_1, \dots, y_m) \simeq \varphi_e^{m+n}(x_1, \dots, x_n, y_1, \dots, y_m)$$

Definition

let W_i denote the domain of the functions φ_i^1

Lemma

the following list contains every recursive enumerable set

$$W_0, W_1, W_2, W_3, W_4, \dots$$

Proposition

define

$$J = \{x \mid x \notin W_x\} \quad K = \{x \mid x \in W_x\}$$

- J is not recursive enumerable
- K is recursive enumerable, but not recursive

Rice's Theorem

Definition

A is (many-one) **reducible** to B (denoted $A \leq_m B$) if \exists total recursive $f: A \rightarrow B$ such that

$$x \in A \quad \text{if and only if} \quad f(x) \in B$$

Definition

$A \subseteq \mathbb{N}$ is called **index set** if

$$x \in A \quad \text{and} \quad \varphi_x^1 \simeq \varphi_y^1 \quad \text{then} \quad y \in A$$

Theorem

let A be an index set; if A is neither \emptyset or \mathbb{N} , then A is not recursive

Rice

Example

the following sets are non-recursive

$$ID = \{x \mid \varphi_x^1 \text{ is the identity}\} \quad REC = \{x \mid W_x \text{ is recursive}\}$$

Proof

consider a partial recursive function φ_c^1 that is undefined everywhere

Claim

if $c \in A$, then $K \leq_m \sim A$

- \exists partial recursive h_K such that

$$h_K(x) = \begin{cases} 1 & x \in K \\ \text{undefined} & \text{otherwise} \end{cases}$$

- observe $\exists e \in \sim A$ and define

$$g(x, y) = \begin{cases} \varphi_e(y) & h_K(x) = 1 \\ \text{undefined} & \text{otherwise} \end{cases}$$

- \exists primitive recursive f , such that $\varphi_{f(x)}(y) \simeq g(x, y)$
- using the fact that A is an index set we obtain $x \in K$ if and only if $f(x) \in \sim A$

hence A is not recursive ■

Content

introduction, propositional logic, semantics, formal proofs, resolution (propositional)

first-order logic, semantics, structures, theories and models, formal proofs, Herbrand theory, completeness of first-order logic, properties of first-order logic, resolution (first-order)

introduction to computability, **introduction to complexity**, finite model theory

beyond first order: modal logics in a general setting, higher-order logics, introduction to Isabelle

The Arithmetical Hierarchy

Definition

a \mathcal{V}_{ar} -formula φ is

$$\Pi_1 \text{ if } \varphi \equiv \forall y \psi(\vec{x}, y) \quad \Sigma_1 \text{ if } \varphi \equiv \exists y \psi(\vec{x}, y) \quad \psi \in \Delta_0$$

Lemma

$$\Sigma_1 = \{\text{r.e. sets}\}$$

Definition

a \mathcal{V}_{ar} -formula φ is

- Π_{n+1} if $\varphi \equiv \forall y \psi(\vec{x}, y)$
- Σ_{n+1} if $\varphi \equiv \exists y \psi(\vec{x}, y)$
- Δ_{n+1} if $\varphi \in \Pi_{n+1} \cap \Sigma_{n+1}$

$\psi \in \Delta_n$

Lemma

$$\Delta_1 = \{\text{recursive sets}\}$$

Complexity Theory

Complexity Theory focuses on the distinction between those problems that are decidable and those that are *really* decidable. There is nothing in this definition of decidability that requires the algorithm to be practical. A decision problem is said to be *feasible* if it can be resolved by an algorithm using a reasonable amount of time and space.

recall

- a Turing machine (TM) is **deterministic** if the transition relation Δ is a function, otherwise it is **nondeterministic**
- $T: \mathbb{N} \rightarrow \mathbb{N}$, $S: \mathbb{N} \rightarrow \mathbb{N}$ denote numeric functions

Definition

time-bounded

- nondeterministic TM **runs in time $T(n)$** or
- TM is **$T(n)$ time-bounded**
- if on all but finitely many inputs x , no computation path takes more than $T(|x|)$ steps before **halting**

Definition

space-bounded

- nondeterministic TM **runs in space $S(n)$** or
- TM is **$S(n)$ space-bounded**
- if on all but finitely many inputs x , no computation path uses more than $S(|x|)$ **worktape** cells

DTIME $(T(n)) := \{L(M) \mid M \text{ is a deterministic multitape TM running in time } O(T(n))\}$

NTIME $(T(n)) := \{L(M) \mid M \text{ is a nondeterministic multitape TM running in time } O(T(n))\}$

DSPACE $(S(n)) := \{L(M) \mid M \text{ is a deterministic multitape TM running in space } O(S(n))\}$

NSPACE $(S(n)) := \{L(M) \mid M \text{ is a nondeterministic multitape TM running in space } O(S(n))\}$

Definition

LOGSPACE := DSPACE($\log n$) **NP** := NTIME($n^{O(1)}$)

NLOGSPACE := NSPACE($\log n$) **PSPACE** := DSPACE($n^{O(1)}$)

P := DTIME($n^{O(1)}$) **NPSPACE** := NSPACE($n^{O(1)}$)

Theorem

LOGSPACE \subseteq **NLOGSPACE** \subseteq **P** \subseteq **NP** \subseteq **PSPACE** \subseteq **NPSPACE**

Problem

MAZE

given

- directed graph G with nodes V
- nodes $s, t \in V$

\exists path between s and t ?

Lemma

MAZE \in NLOGSPACE

Problem

CLIQUE

given

- undirected graph G
- number k

\exists k -clique as a subgraph of G ?

Lemma

CLIQUE \in NP