# *Логика предикатов* *лекция 9*

Лев Дмитриевич Беклемишев
http://lpcs.math.msu.su/vml2010

lbekl@yandex.ru

8.04.2010

# Элементарные теории

Пусть $St_\Sigma$ — множество предложений сигнатуры $\Sigma$

*Опр.*

*Элементарная теория модели* $M$ есть множество $Th(M) \rightleftharpoons \{A \in St_\Sigma : M \models A\}$.

Пример: $Th(\mathbb{N})$

# Элементарная эквивалентность

*Опр.*

Модели $M$ и $N$ сигнатуры $\Sigma$ *элементарно эквивалентны* ($M \equiv N$), если в $M$ и в $N$ истинны одни и те же предложения $\Sigma$, т.е. если $Th(M) = Th(N)$.

*Утверждение.*

*$M \cong N$ влечёт $M \equiv N$. Обратное, вообще говоря, неверно.*

# *Подмодели*

*Опр.*

$(N;\Sigma)$ есть *подмодель* модели $(M;\Sigma)$, если $N \subseteq M$ и для всех $P \in \mathrm{Pred}_\Sigma$, $f \in \mathrm{Func}_\Sigma$, $c \in \mathrm{Const}_\Sigma$ имеем $P_N = P_M \restriction N$, $c_M \in N$, $N$ замкнуто относительно $f_M$ и $f_N = f_M \restriction N$.

*Пример.*

Если $(G;\cdot,1,x^{-1})$ — группа, то подмодели $G$ суть подгруппы группы $G$. Если же $G$ рассматривается как модель $(G;\cdot,1)$, то её подмоделями будут подполугруппы с единицей группы $G$.

# Элементарные подмодели

*Опр.*

Подмодель $(N;\Sigma)$ модели $(M;\Sigma)$ *элементарна* (обозначение $N \preccurlyeq M$), если для всех $A \in \mathrm{Fm}_\Sigma$

$$\forall \vec{x} \in N\,(N \vDash A[\vec{x}] \iff M \vDash A[\vec{x}]).$$

*Утверждение.*

$N \preccurlyeq M$ *влечёт* $N \equiv M$.

*Пример.*

Если $M$ — нестандартная модель $Th(\mathbb{N})$, то $\mathbb{N} \preccurlyeq M$.

# *Теорема Лёвенгейма–Сколема*

Пусть $\Sigma$ — счётная сигнатура.

*Теорема.*

*Всякая модель* $(M; \Sigma)$ *имеет (конечную или) счётную элементарную подмодель.*

*Следствие.*

Всякая непротиворечивая теория в счётной сигнатуре имеет (конечную или) счётную модель.

# Следствия

- Существуют счётные модели $Th(\mathbb{R})$ и $Th(\mathbb{C})$.
- Существует счётная модель элементарной геометрии.
- Если теория множеств $ZFC$ непротиворечива, то существует и счётная модель $ZFC$.

*Доказательство.*

Построим последовательность счётных подмножеств $M$ $\quad N_0 \subseteq N_1 \subseteq N_2 \subseteq \ldots$ такую, что

- $N_0$ — непустое счётное подмножество $M$.
- Для каждой формулы $A[a, \vec{b}]$ и набора $\vec{y} \in N_k$, если $M \vDash \exists v A[v, \vec{y}]$ выберем $x \in M$ такой, что $M \vDash A[x, \vec{y}]$. Добавим все такие $x$ к $N_k$ и получим $N_{k+1}$.

Положим $N \rightleftharpoons \bigcup_{k \geq 0} N_k$.

*Лемма.*

Для любой формулы $A$ и всех $\vec{y} \in N$

$$M \vDash \exists v A[v, \vec{y}] \iff \exists x \in N \;\; M \vDash A[x, \vec{y}].$$

*Лемма.*

$N$ есть подмодель $M$.

*Доказательство.*

Пусть $\vec{x} \in N$, $f \in \mathrm{Func}_{\Sigma}$. Поскольку $M \vDash \exists v\, f(\vec{x}) = v$, имеем $y \in N$ такой, что $M \vDash f(\vec{x}) = y$, т.е. $f_M(\vec{x}) \in N$.

Индукцией по построению $A$ теперь покажем

$$\forall \vec{y} \in N\, (N \vDash A[\vec{y}] \iff M \vDash A[\vec{y}]).$$

- Для атомарных формул $A$ следует из того, что $N$ — подмодель $M$.
- Для $A = \neg B,\ B \wedge C,\ B \vee C$ вытекает из предположения индукции.
- Допустим $A = \exists v B[a/v]$. Тогда

$$M \vDash \exists v B[a/v, \vec{y}] \iff \exists x \in N\ \ M \vDash B[x, \vec{y}]$$
$$\iff \exists x \in N\ \ N \vDash B[x, \vec{y}] \iff N \vDash \exists v B[a/v, \vec{y}].$$

# Обобщение о понижении мощности

*Теорема.*

*Пусть* $(M;\Sigma)$ *— бесконечная модель в счётной сигнатуре и* $\lambda \leq |M|$ *— бесконечная мощность. Тогда найдётся подмодель* $N \preccurlyeq M$ *такая, что* $|N| = \lambda$.

*Доказательство.*

Та же конструкция, но начинаем с любого подмножества $N_0 \subseteq M$ мощности $\lambda$.

# Теорема Мальцева о повышении мощности

Пусть $\Sigma$ — счётная сигнатура.

*Теорема.*

*Для любой бесконечной модели* $(M; \Sigma)$ *и мощности* $\lambda \geq |M|$ *найдётся модель* $(N; \Sigma)$ *такая, что* $M \preccurlyeq N$ *и* $|N| = \lambda$.

*Доказательство.*

Возьмём $X \supseteq M$, $|X| = \lambda$. Рассмотрим сигнатуру $\Sigma_X \rightleftharpoons \Sigma \cup \{\underline{c} : c \in X\}$ и теорию $T := Th(M; \Sigma_X) \cup \{\underline{c} \neq \underline{d} : c, d \in X, c \neq d\}$.

Каждая конечная подтеория $T$ совместна. По теореме о компактности $T$ имеет нормальную модель $N$. Но функция $\varphi : c \longmapsto (\underline{c})_N$ инъективна в силу аксиом $T$, следовательно $|N| \geq |X| = \lambda$. Т.к. $N \models Th(M; \Sigma_X)$, то $\varphi(M)$ есть подмодель $N$, изоморфная $M$ и $\varphi(M) \preccurlyeq N$.

*Следствие.*

Если теория $T$ имеет бесконечную модель, то $T$ имеет модели любой бесконечной мощности.

*Следствие.*

Множество $Th(\mathbb{N})$ всех предложений истинных в стандартной модели арифметики имеет модели любой бесконечной мощности.

# Полные теории

*Опр.*

Теория $T$ *полна*, если

- $T$ непротиворечива;
- Для любого предложения $A$ в языке $T$
  $T \vdash A$ или $T \vdash \neg A$.

*Пример.*

$Th(M)$ для любой модели $M$.

*Утверждение.*

Если $T$ полна и $M \models T$, то $T \equiv Th(M)$.

*Теорема. (Линденбаум)*

Всякая непротиворечивая теория имеет полное расширение.

# *Аксиоматизируемость*

*Опр.*

Теория $T$ *эффективно аксиоматизируема*, если существует алгоритм, распознающий аксиомы $T$.

*Опр.*

Теория $T$ *разрешима*, если существует алгоритм, распознающий теоремы $T$.

*Теорема.*

*Если $T$ полна и эффективно аксиоматизируема, то $T$ разрешима.*

*Доказательство.*

Пусть дано предложение $A$. Перебираем все возможные выводы в $T$ до тех пор, пока не встретим доказательство $A$ или доказательство $\neg A$. Полнота гарантирует, что одно из двух произойдёт.

# Примеры полных эфф. аксиоматизируемых теорий

- Элементарная геометрия (G1-G11). $Th(\mathbb{R}^2; =, \cong, B)$
- Теория вещественно замкнутых упорядоченных полей. $Th(\mathbb{R}; =, <, +, \cdot, 0, 1)$
- Теория алгебраически замкнутых полей характеристики $0$. $Th(\mathbb{C}; =, +, \cdot, 0, 1)$
- Теория плотных линейных порядков без первого и последнего элементов. $Th(\mathbb{Q}; =, <)$

# Универсальные теории

- *Арифметика Пеано* PA.
  Формализует математику конечного.
  Основана на аксиомах для натуральных чисел в сигнатуре $=$, $0$, $S$, $+$, $\cdot$.

- *Теория множеств Цермело–Френкеля (с аксиомой выбора)* ZFC.
  Формализует всю «обычную» математику.
  Основана на аксиомах для множеств и отношения принадлежности $\in$.

# Аксиомы PA

1. аксиомы равенства для $S$, $+$, $\cdot$;
2. $\neg S(a) = 0, \quad S(a) = S(b) \to a = b$,
3. $a + 0 = a, \quad a + S(b) = S(a + b)$,
4. $a \cdot 0 = 0, \quad a \cdot S(b) = a \cdot b + a$,
5. (Схема аксиом индукции)

   $A[a/0] \wedge \forall x\,(A[a/x] \to A[a/S(x)]) \to \forall x\, A[a/x]$
   для любой формулы $A$.

# Первая теорема Гёделя о неполноте

*Теорема.*

*Если $T$ содержит PA, непротиворечива и эффективно аксиоматизируема, то $T$ неполна.*

*Следствие.*

- PA неполна.
- ZFC неполна при условии её непротиворечивости.

# *Вычислимость. Неформальное представление об алгоритмах.*

- *Алгоритм* есть предписание выполнить точно определённую последовательность действий.
- Для данного алгоритма $\mathcal{A}$ определены:
  - *область возможных исходных данных* $X$;
  - *область возможных значений* $Y$.

  В качестве данных обычно рассматриваются слова $X = \Sigma^*$, где $\Sigma$ — конечный алфавит, или числа $X = \mathbb{N}^n$.

# Выполнение алгоритма

- Процесс применения алгоритма $\mathcal{A}$ к данным $x \in X$ происходит по шагам.
- Процесс или заканчивается после конечного числа шагов с результатом $y \in Y$, или останавливается без результата или продолжается бесконечно.
- Таким образом, с алгоритмом $\mathcal{A}$ связывается *частичная функция* $f : X \to Y$.

# *Частичные функции*

*Опр.*

*Частичной функцией* $f : X \to Y$ называется подмножество $f \subseteq X \times Y$ такое, что из $\langle x, y_1 \rangle \in f$ и $\langle x, y_2 \rangle \in f$ следует $y_1 = y_2$.

*Опр.*

Пишем $f(x) = y$ вместо $\langle x, y \rangle \in f$;

$!f(x)$ вместо $\exists y\, f(x) = y$.

*Опр.*

*Областью определения* частичной функции $f$ называется множество
$dom(f) \rightleftharpoons \{x \in X : \exists y \in Y \langle x, y \rangle \in f\}$.

*Опр.*

*Областью значений* частичной функции $f$ называется множество
$rng(f) \rightleftharpoons \{y \in Y : \exists x \in X \langle x, y \rangle \in f\}$.

# *Вычислимые функции*

*Опр.*

Частичная функция $f : X \to Y$ *вычислима*, если она вычисляется некоторым алгоритмом.

В частности, можно говорить о вычислимых функциях $f : \Sigma^* \to \Sigma^*$, $f : \mathbb{N}^k \to \mathbb{N}$ и т.д.

# *Вычислительные модели*

- Машины Тьюринга (А. Тьюринг, Э. Пост)
- Частично рекурсивные функции (К. Гёдель, С. Клини)
- Лямбда-исчисление (А. Чёрч)
- Алгорифмы Маркова
- Машины с неограниченными регистрами
- Pascal, C, Java, Lisp, ...

# Эквивалентность вычислительных моделей

*Теорема.*

*Каждая из вышеперечисленных моделей определяет один и тот же класс вычислимых частичных функций* $f : \Sigma^* \to \Sigma^*$.

# *Тезис Чёрча–Тьюринга*

Тезис: Любая вычислимая в интуитивном смысле частичная функция $f : \Sigma^* \to \Sigma^*$ вычислима на машине Тьюринга.

*Замечание.*
Это утверждение не является математическим, но говорит об адекватности математической модели (вычислимости по Тьюрингу) *реальному* явлению (вычислимости).

# *Подтверждения тезиса Чёрча–Тьюринга*

Все попытки построения более общих вычислительных моделей неизбежно приводили к тому же самому классу вычислимых функций.

# Физический тезис Чёрча–Тьюринга

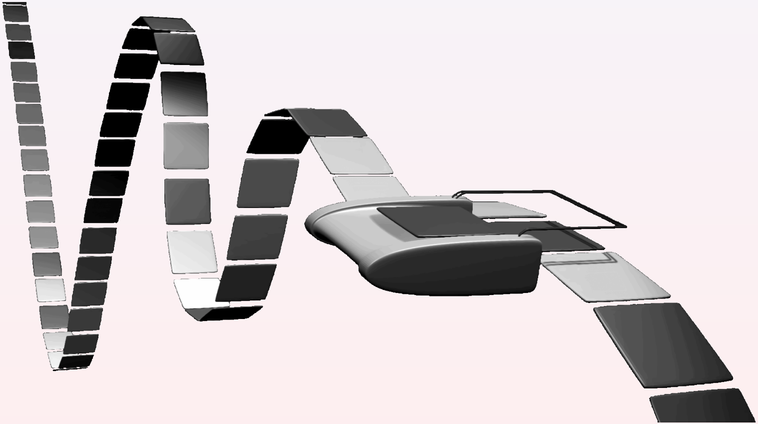
Текущему уровню знаний не противоречит и более сильный

Тезис: Всякая функция $f : \Sigma^* \to \Sigma^*$, вычислимая на (идеализированном) *физически реализуемом* устройстве, вычислима на машине Тьюринга.

*Замечание.*

Физический тезис предполагает возможность аналогового вычисления, квантово–механические эффекты и т.д.

# Машина Тьюринга



# Машины Тьюринга

*Опр.*

Машина Тьюринга задаётся конечными

- рабочим алфавитом $\Sigma$, содержащим символ $\#$ (пробел);
- множеством состояний $Q$, содержащим состояния $q_1$ (начальное) и $q_0$ (конечное);
- набором команд (программой) $P$.

# Команды

- Команды имеют вид $qa \to rb\nu$, где $q, r \in Q$, $a, b \in \Sigma$ и $\nu \in \{L, N, R\}$.

  *«прочтя символ $a$ в состоянии $q$ перейти в состояние $r$, заменить содержимое ячейки на $b$ и сместиться влево (L), остаться на месте (N) или сместиться вправо (R) на одну ячейку, в зависимости от значения $\nu$»*

- Требуется, чтобы в программе $P$ была ровно одна команда с левой частью $qa$ для каждого $q \in Q \setminus \{q_0\}$ и $a \in \Sigma$.

*Соглашение*: команды вида $qa \longrightarrow qaN$, приводящие к зацикливанию, можно не указывать.

*Опр.*

*Машина Тьюринга* есть набор
$M = \langle Q, \Sigma, P, q_0, q_1 \rangle$.

*Пример.*

Пусть $\Sigma = \{\#, 0, 1\}$, $Q = \{q_0, q_1\}$, а $P$ состоит из следующих команд:

$$\begin{aligned} q_1\# &\mapsto q_1\#R \\ q_10 &\mapsto q_11R \\ q_11 &\mapsto q_10R \end{aligned}$$

Что делает эта машина Тьюринга?

Модифицируем программу.

Пусть $\Sigma = \{\#, 0, 1\}$, $Q = \{q_0, q_1, q_2\}$, а $P$ состоит из следующих команд:

$$\begin{array}{rcl}
q_1\# & \mapsto & q_1\#R \\
q_1 0 & \mapsto & q_2 1 R \\
q_1 1 & \mapsto & q_2 0 R \\
q_2 0 & \mapsto & q_2 1 R \\
q_2 1 & \mapsto & q_2 0 R \\
q_2\# & \mapsto & q_0\#N
\end{array}$$