Математическая логика и теория вычислимости
Лекция 14. Рекурсивные функции

Денис Николаевич Москвин

Совместная магистратура JetBrains и ИТМО
Разработка ПО / Software Engineering

26.05.2021

План лекции

План лекции

Примитивно рекурсивные функции

- Базисные функции:
  - константа 0;
  - функция следования: $s(x) = x + 1$;
  - семейство функций проекции: $\pi_k^i(x_1, \ldots, x_k) = x_i$.
- Операции:
  - подстановка

$$h(x_1, \ldots, x_n) = f(g_1(x_1, \ldots, x_n), \ldots, g_k(x_1, \ldots, x_n))$$

  - примитивная рекурсия

$$\begin{array}{lcl} h(x_1, \ldots, x_k, 0) & = & f(x_1, \ldots, x_k) \\ h(x_1, \ldots, x_k, s(y)) & = & g(x_1, \ldots, x_k, y, h(x_1, \ldots, x_k, y)) \end{array}$$

Утверждения с прошлой лекции

- Мы показали, что сложение, умножение, усеченное вычитание примитивно рекурсивны, а также что пересечение, объединение и дополнение примитивно рекурсивных множеств примитивно рекурсивны.
- Ограниченные кванторы, примененные к примитивно рекурсивным предикатам дают примитивно рекурсивные предикаты:

$$\begin{aligned} S(x,z) &= \forall y \leqslant z.\ R(x,y) \\ T(x,z) &= \exists y \leqslant z.\ R(x,y) \end{aligned}$$

Примитивно рекурсивный ограниченный график

- Пусть график функции $f(x)$ примитивно рекурсивен, а ее значения ограничены сверху примитивно рекурсивной $g(x)$. Тогда f примитивно рекурсивна.

$$\Gamma_f = \{(x, f(x)) \mid x \in \mathbb{N}\}$$

- Действительно, рассмотрим характеристическую функцию графика $r(x, y)$, тогда

$$f(x) = \sum_{y=0}^{\infty} r(x, y)y = \sum_{y=0}^{g(x)} r(x, y)y$$

А конечная сумма примитивно рекурсивна.

Ограниченный оператор минимизации

- Введем для предиката $P(x)$ *ограниченный оператор минимизации* $\mu x < z.\ P(x)$. Он возвращает наименьший x строго меньший z, для котрого этот предикат выполняется, или z, если такого x нет.
- Пусть функция $g(x)$ и предикат $P(x, y)$ примитивно рекурсивны. Тогда примитивно рекурсивна функция

$$f(x) = \mu y < g(x).\ P(x, y)$$

- Действительно, график $f(x)$ легко описать ограниченными кванторами

$$\begin{array}{rcl} P(x,y) = 1 & \wedge & \forall z < y\ P(x,z) = 0 \quad \vee \\ y = g(x) & \wedge & \forall z < y\ P(x,z) = 0 \end{array}$$

Примитивно рекурсивная нумерация пар

- Хотим по паре $\langle x, y \rangle$ примитивно рекурсивно задавать ее номер $[x, y]$ и, наоборот, по номеру $[x, y]$ находить значения элементов пары $x = p_1([x, y])$, $y = p_2([x, y])$.

Примитивно рекурсивная нумерация пар

- Хотим по паре $\langle x, y\rangle$ примитивно рекурсивно задавать ее номер $[x, y]$ и, наоборот, по номеру $[x, y]$ находить значения элементов пары $x = p_1([x, y])$, $y = p_2([x, y])$.
- Нумерация «по диагонали» кодируется полиномом

$$[x, y] = ((x + y)^2 + 3x + y)/2$$

- Почему $p_1(n)$ и $p_2(n)$ будут примитивно рекурсивны?

Примитивно рекурсивная нумерация пар

- Хотим по паре $\langle x, y\rangle$ примитивно рекурсивно задавать ее номер $[x, y]$ и, наоборот, по номеру $[x, y]$ находить значения элементов пары $x = p_1([x, y])$, $y = p_2([x, y])$.
- Нумерация «по диагонали» кодируется полиномом

$$[x, y] = ((x + y)^2 + 3x + y)/2$$

- Почему $p_1(n)$ и $p_2(n)$ будут примитивно рекурсивны?

$$\begin{aligned} p_1(n) &= \mu x \leqslant n.\ \exists y \leqslant n.\ [x, y] = n \\ p_2(n) &= \mu y \leqslant n.\ \exists x \leqslant n.\ [x, y] = n \end{aligned}$$

Примитивно рекурсивная нумерация пар

- Хотим по паре $\langle x, y \rangle$ примитивно рекурсивно задавать ее номер $[x, y]$ и, наоборот, по номеру $[x, y]$ находить значения элементов пары $x = p_1([x, y])$, $y = p_2([x, y])$.
- Нумерация «по диагонали» кодируется полиномом

$$[x, y] = ((x + y)^2 + 3x + y)/2$$

- Почему $p_1(n)$ и $p_2(n)$ будут примитивно рекурсивны?

$$\begin{aligned} p_1(n) &= \mu x \leqslant n.\ \exists y \leqslant n.\ [x, y] = n \\ p_2(n) &= \mu y \leqslant n.\ \exists x \leqslant n.\ [x, y] = n \end{aligned}$$

- Есть и другие способы нумерации:

$$\begin{aligned} &[x, y] = (2x + 1)2^y - 1 \\ &[x, y] = 2^x 3^y \\ &\ldots \end{aligned}$$

- Пусть унарные функции f и g заданы равенствами

$$\begin{array}{lcl} f(0) & = & a \\ g(0) & = & b \\ f(s(n)) & = & F(n, f(n), g(n)) \\ g(s(n)) & = & G(n, f(n), g(n)) \end{array}$$

где a и b — числа, а F и G — бинарные примитивно рекурсивные функции. Тогда f и g примитивно рекурсивны.

- **Доказательство.**

$$\begin{array}{lcl} h(0) & = & [a, b] \\ h(s(n)) & = & \big[F\big(n, p_1(h(x)), p_2(h(x))\big), \\ & & \ G\big(n, p_1(h(x)), p_2(h(x))\big)\big] \end{array}$$

$$f(x) = p_1(h(x)), \qquad g(x) = p_2(h(x)). \quad \blacksquare$$

Возвратная рекурсия

- Пусть $g(x)$ и $h(x, y)$ — унарная и бинарная примитивно рекурсивные функции, причем $g(x) < x$ при положительных x. Тогда унарная функция f, заданная равенствами
$$\begin{aligned} f(0) &= a \\ f(n) &= h(n, f(g(n))) \end{aligned}$$
примитивно рекурсивна.
- **Доказательство.** (см. Верещагина и Шеня)
- Мораль: при определении примитивной рекурсии можно использовать не только предыдущее, но и любое меньшее значение.

План лекции

1. Примитивно рекурсивные функции
2. Примитивная рекурсия и машины Тьюринга
3. Частично рекурсивные функции
4. Общерекурсивные функции

Примитивно рекурсивное кодирование машины Тьюринга

- *Конфигурация* машины Тьюринга это:
  - содержимое ленты;
  - положение считывающего устройства;
  - текущее состояние.
- Можем закодировать в виде тройки (l, r, q), где l и r — списки, задающие левую и правую части ленты, а q — номер состояния.
- Если $|\Sigma| = k$, то списки удобно кодировать в системе счисления с основанием k в виде полинома

$$a_0 + a_1 k + a_2 k^2 + \ldots + a_n k^n$$

- При этом стандартные для машины Тьюринга операции легко выражаются через примитивно-рекурсивные $\mathtt{push}$, $\mathtt{pop}$ и $\mathtt{top}$.

Примитивно рекурсивное кодирование машины Тьюринга (2)

- Умеем кодировать конфигурацию машины Тьюринга как тройку (l, r, q), ее кодируем числом $K = [[l, r], q]$.
- Шаг машины Тьюринга кодируем примитивно рекурсивной $K' = \mathtt{Step}(K)$.
- Конфигурацию через n шагов кодируем примитивно рекурсивной $K' = \mathtt{nStep}(K, n)$.
- Построение начальной конфигурации по входу и выхода по финальной конфигурации кодируем примитивно рекурсивными $K = \mathtt{Init}(x)$ и $y = \mathtt{Result}(k)$.

Примитивная рекурсивное кодирование машины Тьюринга (3)

- **Теорема.** Пусть f — вычислимая на машине Тьюринга функция, и число шагов до остановки на входе x не превышает $g(x)$, где $g(x)$ — некоторая примитивно рекурсивная функция. Тогда $f(x)$ — примитивно рекурсивная функция.
- **Доказательство.**

$$f(x) = \mathtt{Result}(\mathtt{nStep}(\mathtt{Init}(x), g(x))) \quad \blacksquare$$

План лекции

- *Частично рекурсивные* функции (partial recursive functions) определяют индуктивно.
- Базисные функции:
  - константа 0;
  - функция следования: $s(x) = x + 1$;
  - семейство функций проекции: $\pi_k^i(x_1, \ldots, x_k) = x_i$.
- Операции:
  - подстановка;
  - примитивная рекурсия;
  - минимизация: если $f(x_1, \ldots, x_k, y)$ — частично рекурсивная функция, то функция

$$g(x_1, \ldots, x_k) = \mu y.\ f(x_1, \ldots, x_k, y) = 0$$

  тоже является частично рекурсивной.

Частично рекурсивные функции: примеры

- Частично рекурсивные функции могут быть не всюду определены.
- **Пример.** Рассмотрим $f(x, y) = s(x) + y$. Где определена

$$g(x) = \mu y.\ f(x, y) = 0$$

Частично рекурсивные функции: примеры

- Частично рекурсивные функции могут быть не всюду определены.
- **Пример.** Рассмотрим $f(x, y) = \mathbf{s}(x) + y$. Где определена

$$g(x) = \mu y.\ f(x, y) = 0$$

- **Пример.** Рассмотрим версию вычитания $x - y$, не определенную при $y > x$. Покажите, что она частично рекурсивна.

- Частично рекурсивные функции могут быть не всюду определены.
- **Пример.** Рассмотрим $f(x, y) = \mathbf{s}(x) + y$. Где определена

$$g(x) = \mu y.\ f(x, y) = 0$$

- **Пример.** Рассмотрим версию вычитания $x - y$, не определенную при $y > x$. Покажите, что она частично рекурсивна.

$$x - y = \mu z.\ |x - (y + z)| = 0$$

(модуль разности примитивно рекурсивен.)

- Частично рекурсивные функции могут быть не всюду определены.
- **Пример.** Рассмотрим $f(x, y) = s(x) + y$. Где определена

$$g(x) = \mu y.\ f(x, y) = 0$$

- **Пример.** Рассмотрим версию вычитания $x - y$, не определенную при $y > x$. Покажите, что она частично рекурсивна.

$$x - y = \mu z.\ |x - (y + z)| = 0$$

(модуль разности примитивно рекурсивен.)

- Если частично рекурсивная функция определена всюду, ее называют *общерекурсивной* (total recursive functions).

Вычислимость частично рекурсивных функций

- **Теорема.** Если f — частично рекурсивная функция, то она вычислима на машине Тьюринга.
- **Доказательство (набросок).** Подстановка соответствует последовательному вычислению, оператор примитивной рекурсии — циклу `for`, оператор минимизации — циклу `while`.
  Все эти операции несложно реализовать на машине Тьюринга. ■

Частичная рекурсивность вычислимых функций

- **Теорема.** Если f вычислима на машине Тьюринга, то она частично рекурсивна.
- **Доказательство.** Пусть M — машина Тьюринга, вычисляющая f. Рассмотрим предикат

  $T(x, y, t) =$«M на x ввыдает y не более чем за t шагов»

  Этот предикат примитивно рекурсивен. Сделаем из y и t пару $z = [y, t]$ (какой-нибудь нумерацией) и рассмотрим

  $$T'(x, z) = T(x, p_1(z), p_2(z))$$

  Тогда

  $$f(x) = p_2(\mu z.\ T'(x, z)) \quad \blacksquare$$

Теорема Клини о нормальной форме

- **Теорема (Клини).** Всякая частично рекурсивная функция представима в виде

$$f(x) = g(\mu z.\ h(x, z) = 0)$$

где g и h — некоторые примитивно рекурсивные функции.
- **Доказательство.** Берем машину Тьюринга, вычисляющую f, и пользуемся предыдущей теоремой. ■
- **Следствие.** Всякое перечислимое множество есть проекция примитивно рекурсивного.
- **Доказательство.** Перечислимое множество есть область определения вычислимой функции, берем ее нормальную форму и обнаруживаем, что область определения — проекция $\{\langle x, z\rangle \mid h(x, z) = 0\}$. ■

План лекции

1. Примитивно рекурсивные функции
2. Примитивная рекурсия и машины Тьюринга
3. Частично рекурсивные функции
4. Общерекурсивные функции

- Напомним, что *общерекурсивной* называют всюду определенную частично рекурсивную функцию.
- Бывают ли функции общерекурсивные, но не примитивно рекурсивные?

Общерекурсивные функции

- Напомним, что *общерекурсивной* называют всюду определенную частично рекурсивную функцию.
- Бывают ли функции общерекурсивные, но не примитивно рекурсивные? Да.
- Перенумеруем все унарные примитивно рекурсивные функции (это можно сделать, поскольку описание примитивно рекурсивной функции можно задать словом в некотором алфавите).
- Рассмотрим функцию $U(n, x)$, которая паре (n, x) ставит в соответствие функцию с номером n, вычисленную на аргументе x. Это будет универсальная функция для класса примитивно рекурсивных.
- Применим теперь диагонализацию, рассмотрев

$$d(n) = U(n, n) + 1$$

Получим всюду определенную функцию, отличную от любой примитивно рекурсивной.

Функция Аккермана (Wilhelm Ackermann)

- Рассмотрим семейство функций

$$\begin{aligned} \alpha_0(x) &= x+1 \\ \alpha_{i+1}(x) &= \alpha_i^{[x+2]}(x) \end{aligned}$$

где $f^{[n]}(x)$ — это n-кратное применение f, то есть $f(f(\dots f(x)\dots))$.

- Очевидные свойства функции $\alpha_i(x)$:
  - $\forall i, x.\ \alpha_i(x) > x$;
  - $x_2 > x_1 \to \alpha_i(x_2) > \alpha_i(x_1)$;
  - $i > j \to \alpha_i(x) > \alpha_j(x)$;
  - $\alpha_{i+1}(x) \geqslant \alpha_i(\alpha_i(x))$.

- **Теорема.** Любую примитивно рекурсивную функцию $f : \mathbb{N}^n \to \mathbb{N}$ можно оценить сверху функцией $\alpha_k(x)$ с фиксированным k:

$$\forall f.\ \exists k.\ \forall \overrightarrow{x} \in \mathbb{N}^n.\ f(\overrightarrow{x}) \leqslant \alpha_k(\max \overrightarrow{x}).$$

- **Доказательство.** (Индукция по построению примитивно рекурсивной функции).
  Базисные
  - $0 \leqslant \alpha_0(x)$;
  - $s(x) = \alpha_0(x)$;
  - $\pi_n^i(\overrightarrow{x}) = x_i \leqslant \max \overrightarrow{x} \leqslant \alpha_0(\max \overrightarrow{x})$.

Подстановка.

$$h(\overrightarrow{x}) = f(g_1(\overrightarrow{x}), \dots, g_m(\overrightarrow{x}))$$

причем по IH имеется k, такое что $f(\overrightarrow{y}) \leqslant \alpha_k(\max \overrightarrow{y})$ и для любого $j \leqslant m$ верно $g_j(\overrightarrow{x}) \leqslant \alpha_k(\max \overrightarrow{x})$. Тогда

$$h(\overrightarrow{x}) \leqslant \alpha_k(\max(g_1(\overrightarrow{x}), \dots, g_m(\overrightarrow{x}))) \leqslant \alpha_k(\alpha_k(\overrightarrow{x})) \leqslant \alpha_{k+1}(\overrightarrow{x})$$

Примитивная рекурсия.

$$\begin{array}{rcl} h(\overrightarrow{x}, 0) & = & f(\overrightarrow{x}) \\ h(\overrightarrow{x}, s(y)) & = & g(\overrightarrow{x}, y, h(\overrightarrow{x}, y)) \end{array}$$

причем по IH имеется k, такое что $f(\overrightarrow{x}) \leqslant \alpha_k(\max \overrightarrow{x})$ и $g(\overrightarrow{z}) \leqslant \alpha_k(\max \overrightarrow{z})$. Тогда

$$\begin{array}{rcl} h(\overrightarrow{x}, 0) & \leqslant & \alpha_k(\max \overrightarrow{x}) \\ h(\overrightarrow{x}, 1) & = & g(\overrightarrow{x}, 0, h(\overrightarrow{x}, 0)) \leqslant \alpha_k(\max(\overrightarrow{x}, 0, \alpha_k(\max \overrightarrow{x}))) \leqslant \\ & & \leqslant \alpha_k(\alpha_k(\max \overrightarrow{x})) \\ h(\overrightarrow{x}, 2) & \leqslant & \alpha_k(\alpha_k(\alpha_k(\max \overrightarrow{x}))) \\ & \dots & \\ h(\overrightarrow{x}, i) & \leqslant & \alpha_k^{[i+1]}(\max \overrightarrow{x}) \leqslant \alpha_{k+1}(\max(\overrightarrow{x}, i)) \quad \blacksquare \end{array}$$

- **Следствие 1.** Примитивно рекурсивная функция, содержащая k операторов примитивной рекурсии и подстановки растет не быстрее α_{k+1}.
- **Следствие 2.** Функция $A(n) = \alpha_n(n)$ растет быстрее любой примитивно рекурсивной функции.
- $A(n)$ — конструктивный пример общерекурсивной функции, которая не является примитивно рекурсивной.