Логика

Мы

6 июля 2023 г.

Оглавление

Глава 1

Аксиоматический метод

Базовое понятие - это неопределяемое понятие.

1.1

Зафиксировать базовые понятия.

1.2

Зафиксировать аксиомы, связывающие понятия.

1.3

Выводить следствия по правилам логики.

Глава 2

Теория множеств (ZFC)

2.1 Базовые понятия

Принадлежность ($x \in y$).

2.2 Аксиомы

2.2.1 Равенства

$$x = y \; \forall z \; (x \in z \iff y \in z)$$

2.2.2 Пары

$$\exists \{x, y\}$$

2.2.3 Объединения

$$\exists \cup x$$

2.2.4 Степени

$$\exists \mathcal{P}(x)$$

2.2.5 Выделения

$$\{x \in A \mid \varphi(x)\}$$

2.2.6 Бесконечности

$\exists S -$ индуктивное множество

2.2.7 Выбора

$$\varnothing \notin S \; \exists f \begin{cases} f : S \to \cup S \\ \forall s \in S \; f(s) \in s \end{cases}$$

2.2.8 Регулярности (фундированности)

Необязательная аксиома.

$$\exists y \in x \; \forall z \in x \; z \notin y$$

2.2.9 Подстановки

Не знаю, что она означает. Необязательная аксиома. Аксиома выделения - это часть данной аксиомы.

$$\forall x \; \exists! y \; \varphi(x, y) \implies \forall X \; \exists z \; \forall u \; (u \in z \iff \exists x \in X \; \varphi(x, u))$$

2.3 Определения

Класс $= \{x \mid \varphi(x)\}$. Не все классы являются множествами. Все множества являются классами.

$$x = y \iff (z \in x \iff z \in y)$$
$$y = \{\ldots, x, \ldots\} \iff x \in y$$
$$x \subseteq y \iff \forall z \in x \; z \in y$$
$$x \subsetneq y \iff \begin{cases} x \neq y \\ x \subseteq y \end{cases}$$
$$\varnothing - \text{пустое множество}$$
$$x \notin \varnothing$$
$$y \in \mathcal{P}(x) \iff y \subseteq x$$
$$y \in \cup x \iff \exists z \begin{cases} z \in x \\ y \in z \end{cases}$$
$$x - \text{транзитивное множество} \iff \cup x \subseteq x$$
$$\cap x = \{y \in \cup x \mid z \in x \; y \in z\}$$
$$a \cup b = \cup \{a, b\}$$
$$a \cap b = \{x \in a \mid x \in b\}$$
$$a \setminus b = \{x \in a \mid x \notin b\}$$
$$a \triangle b = (a \setminus b) \cup (b \setminus a)$$
$$S - \text{индуктивное множество} \iff \begin{cases} \varnothing \in S \\ \forall s \in S \; s \cup \{s\} \in S \end{cases}$$

2.4 Теоремы

Глава 3

Формальные языки

$a \neq \varnothing \iff a -$ алфавит
$a \in A \iff a -$ символ (буква)
$f : \underline{n} \to A \iff f -$ слово
$\varepsilon -$ пустое слово
$\varepsilon = \varnothing$

Глава 4

Программная инженерия

Любую программу можно написать на низкоуровневом языке - языке, наиболее приближённом к устройству компьютера. Однако, тем не менее, разрабатывают всё новые языки программирования. Далее рассматриваются причины, по которым возникает нужда в высокоуровневых языках.

4.1 Изменяемость

Программистам в течение жизненного цикла разработки ПО приходится изменять программу. По причине изменений требований к продукту или для устранения ошибки.

Чтобы что-то изменить, нужно найти всю имплементацию этого чего-то в программе. С этим возникают две трудности: имплементация этого чего-то простирается в большой части кода - слишком много приходится править, чтобы внести нужное изменение - и код трудно читаем - сложно понять, где то, что нам нужно.

Языки программирования создают такими, чтобы они как можно более полно решали данные проблемы.

Часто добавляют "мультипарадигменные" конструкции, которые должны матчаться в нашем мозгу с устоявшимися паттернами. Однако всевозможных паттернов настолько много, что данные вводящиеся конструкции только капля в море.

Вторым способом бороться с данными проблемами, который работает всегда, является продолжающийся рефакторинг согласно сложности Джона.

Глава 5

Трансляторы

Компилятор - это программа, переводящая текст программы с одного языка на другой.

Интерпретатор - это программа, выполняющая код программы, не переводя её на другой язык.

Компоновщик (линкер) - это программа, выполняющая разрешение внешних адресов памяти, по которым код из одного файла может обращаться к информации из другого файла.

Загрузчик - это программа, которая помещает все выполнимые объектные файлы в память для выполнения.

Компиляция состоит из анализа и синтеза.

В течение компиляции код может переводиться по цепочке в несколько промежуточных представлений.

Таблица символов содержит в себе информацию, которая накапливается на протяжении компиляции.

Проход (pass) - это этапы компиляции, преобразующие один файл в другой (необязательно в файл с целевым кодом).

5.1 Этапы компиляции

5.1.1 Лексический анализ (сканирование)

Лексема - это значащая последовательность символов кода.

Токен - это значение <имя токена, значение атрибута>, представляющее лексему, где значение атрибута указывает на запись в таблице символов.

5.1.2 Синтаксический анализ (парсинг/разбор)

Синтаксический анализатор структурирует токены в синтаксическое дерево.

Контекстно-свободная грамматика (КС-грамматика).

Терминальный символ - это элементарный символ языка, определяемый грамматикой.

Нетерминальный символ - это множество строк терминалов, заданное продукцией.

Продукция - это определение конкретного нетерминального символа. Записывается как $a \to b$, где a - нетерминал, называемый заголовком (левой частью) продукции, b - последовательность (декартово произведение, если про множества) нетерминалов и/или (объединение, если про множества) терминалов (последовательность может быть пустой, что соответствует пустой строке или пустому множеству), называемая телом (правой частью) продукции.

Контекстно-свободная грамматика имеет четыре компонента:

1. Множество терминальных символов.
2. Множество нетерминнальных символов.
3. Множество продукций.
4. Стартовый нетерминальный символ.

Грамматика выводит (порождает) строки, начиная со стартового символа.

Язык - это множество строк терминалов, определяемые грамматикой.

Синтаксический анализ - это выяснение для строки терминалов способа её вывода из стартового символа грамматики.

Дерево разбора - это дерево, представляющее порождение конкретной строки языка.

Неоднозначная грамматика - это грамматика, имеющая более одного дерева разбора для какой-то строки.

Форма Бэкуса-Наура (BNF) - это другая форма записи КС грамматики.

Для любой КС грамматики существует парсер, который требует для разбора строки из n терминалов время, не превышающее $O\left(n^3\right)$.

Вручную обычно используют нисходящий разбор. Восходящий разбор работает в большем количестве случаем, чем нисходящий, поэтому его используют в автоматическом построении парсеров.

Синтаксически управляемая трансляция - это трансляция, выполняемая путём присоединения правил или программных фрагментов к продукциям грамматики.

Атрибут - это некоторая величина, связанная с программной конструкцией.

Синтаксически управляемое определение связывается:

1. С каждым грамматическим символом множеством атрибутов.

2. С каждой продукцией множеством семантических правил для вычисления значений атрибутов, связанных с символами продукции.

Простое синтаксически управляемое определение - это синтаксически управляемое определение, в котором атрибуты идут в том же порядке, что и соответствующие терминалы и нетерминалы.

Синтезированный атрибут - это атрибут узла дерева разбора, значение которого определяется на основании атрибутов дочерних узлов этого узла и атрибутов самого этого узла.

Аннотированное дерево разбора - это дерево разбора с указанием значений атрибутов в каждом узле.

(Синтаксически управляемая) схема трансляции - это запись присоединённых к продукциям грамматики программных фрагментов.

5.1.3 Семантический анализ

Семантический анализатор проверяет синтаксическое дерево на корректность.

5.1.4 Генерация промежуточного кода

Генерация кода для абстрактной вычислительной машины.

5.1.5 Оптимизация кода

Оптимизация промежуточного кода.

5.1.6 Генерация кода

Генерация кода на целевом языке.

Глава 6

Необработанное

Автонимный способ обозначения - это способ обозначения, при котором формальные выражения обозначаются так же, как и их значения.

Высказывательная форма.

Именная форма - это выражение с переменной.

Связанные переменные - это переменные, вместо которых нельзя подставить значение.

Основания математики - это раздел (в книге сказано "аспект") математической логики, изучающий объекты математики, истинные свойства этих объектов, на основании которых можно вести рассуждения, а также "сохраняющие истину" способы рассуждений.