

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМ. Н.И. ЛОБАЧЕВСКОГО
ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МАТЕМАТИКИ И МЕХАНИКИ

Учебный курс

МЕТОДЫ ПРОГРАММИРОВАНИЯ - 2





Нижегородский государственный университет им. Н.И. Лобачевского
Институт информационных технологий, математики и механики

Учебный курс:

Методы программирования - 2

Тема 1:

Структуры действия и структуры данных

Гергель В.П., профессор ,
директор института ИТММ

Глава 1.

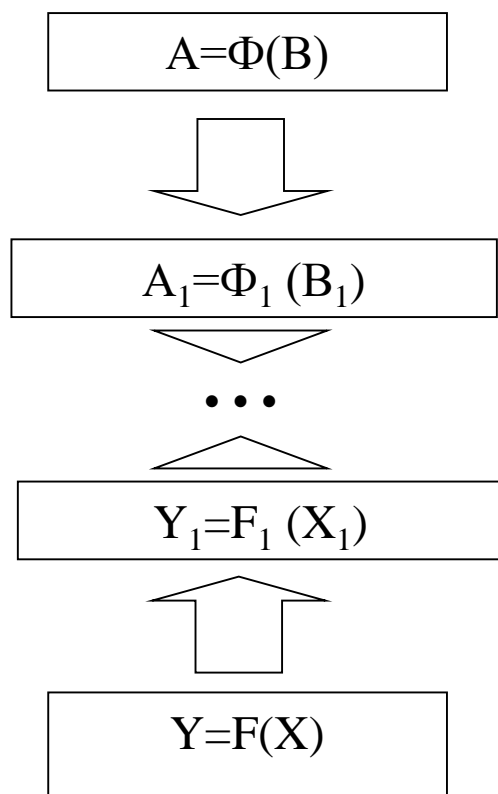
Структура действия и структуры данных

1.1. Структуры данных

1. Структуры данных, порождаемые структурой действия
2. Структуры данных, для которых возможны рекурсивные вычисления
3. Понятие структуры данных
4. Схема и экземпляр структуры данных
5. Именованное представление элементов структуры

Вопросы для обсуждения

Общая схема отображения математических моделей на ЭВМ ...




(**A** – выходные данные, **B** – исходные данные, **Φ** – правило преобразования)

При отображении модели на ЭВМ данные модели и необходимые преобразования реализуются при помощи уже существующих объектов на ЭВМ

Таких уровней декомпозиции может быть несколько

Подобная ситуация прослеживается и при разработке программ для автоматизации деятельности ЭВМ (реализация уровней программного обеспечения – снизу вверх)

Общая схема отображения математических моделей на ЭВМ ...

 Отображение математических моделей на аппаратуру ЭВМ можно себе представить как последовательность этапов построения иерархически-согласованных моделей

Сверху вниз - этапы построения всё более конкретных и детальных моделей, ориентированных на отображение на аппаратуру ЭВМ

Снизу вверх - этапы построения всё более общих моделей, более приближённых к объектам исследования

Общая схема отображения математических моделей на ЭВМ

Как правило, между моделями верхнего и нижнего уровня остаётся несколько нереализованных промежуточных слоёв и ликвидация этого разрыва и есть *основная задача программиста*.

Общий аппарат для построения программных систем -
Структуры данных и структуры действий

1.1. Структуры данных ...

1. Структуры данных, порождаемые структурой действия

ЭВМ является универсальной, поскольку все операторы любого алгоритма разлагаются в последовательность базовых операций. Разложение оператора рождает разложение операнда

Пример: Скалярное произведение

$$(a,b) = a_1 \cdot b_1 + a_2 \cdot b_2 + a_3 \cdot b_3$$

Как располагать значения ?

$$a = (a_1, a_2, a_3) \text{ или } a = (a_3, a_1, a_2)$$

Структура операндов определяется структурой действия



1.1. Структуры данных ...

2. Структуры данных, для которых возможны рекурсивные вычисления ...

Запишем алгоритм скалярного произведения для произвольной длины вектора

$$\sum_0 = 0 \dots \sum_i = \sum_{i-1} + a_i * b_i$$

Результат шага даётся через результат предшествующего шага. Такое описание называется *рекурсией*. Даёт краткое описание процесса.

Индексы можно убрать в Σ , если используется значение только последней частичной суммы.

$$\Sigma = 0 \dots \Sigma = \Sigma + a_i * b_i$$

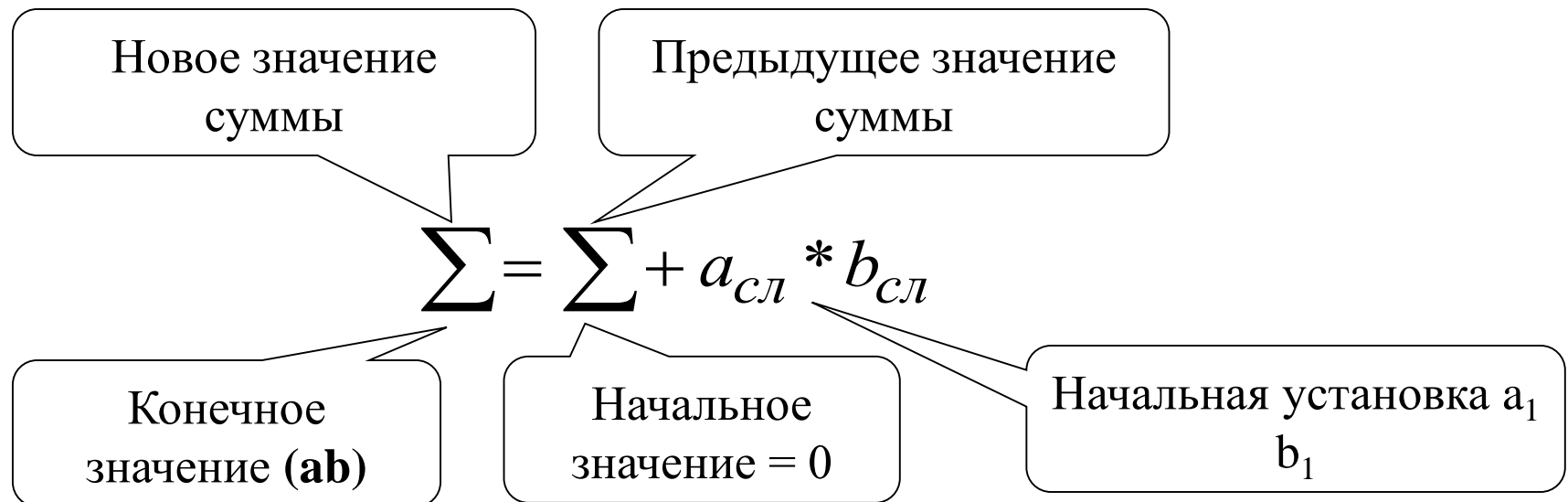


1.1. Структуры данных ...

2. Структуры данных, для которых возможны рекурсивные вычисления ...

Пусть:

- Элементы векторов располагаются последовательно слева направо,
- Имеется указатель текущего элемента,
- Определена операция "следующий элемент"



1.1. Структуры данных ...

2. Структуры данных, для которых возможны рекурсивные вычисления ...

📖 Введение рекурсии (реализация циклов) требует установления *отношения следования* между элементами данных (т.е. введение понятия соседства и перехода к следующему).

☑ Использование рекурсивно (итеративно) описанного действия предполагает наличие структуры операндов и эта структура является **свойством операндов**.

1.1. Структуры данных ...

3. Понятие структуры данных ...

Одной из наиболее общих математических абстракций является понятие алгебраической системы $\langle A, O, R \rangle$, где

A - множество операндов $A^{n_i} \rightarrow A, i \in I$

O - множество операций $r_j \subseteq A^{m_j}, j \in J$

R - множество отношений

n_i - арность операций, m_j - арность отношений

Если операций нет *модель (структура)*

Если отношений нет *универсальная алгебра*



1.1. Структуры данных ...

3. Понятие структуры данных ...

Определение 1.1. *Математическая структура*

$$S = (M_1, \dots, M_k; p_1, \dots, p_s)$$

есть одно или несколько множеств M_1, \dots, M_k , элементы которых находятся в некоторых отношениях p_1, \dots, p_s .

M_1, \dots, M_k – базисные множества структуры.

Каждое отношение p_i есть двоичная функция, аргументами которой являются элементы базисного множества. Если аргументы функции находятся в отношении, то значение $p_i = \text{ИСТИНА}$.

Определение 1.2. *Структура данных* есть модель данных в виде математической структуры.



1.1. Структуры данных ...

3. Понятие структуры данных ...

Примеры

- множество операндов и операций и порядок их записи (*арифметическое выражение*),
- множество узлов детали и порядок их соединения (*чертеж*),
- множество людей и родственные связи между ними (*генеалогическое дерево*),
- множество населенных пунктов и пути сообщения (*карта*)

1.1. Структуры данных ...

3. Понятие структуры данных ...

Пример 1.1. Вектор $a = (a_1, \dots, a_n)$

$$M_a = \{a_1, \dots, a_n\}$$

$$p_a \{a_i, a_j\}_{1 \leq i, j \leq n} = \begin{cases} u, & j = i + 1 \\ л, & j \neq i + 1 \end{cases}$$

Модель вектора (структура данных)

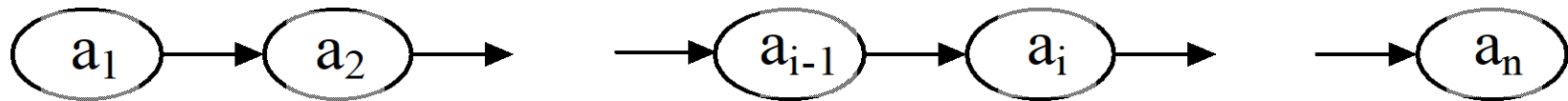
$$S_a = (M_a, p_a)$$



1.1. Структуры данных ...

3. Понятие структуры данных ...

- ✓ Структуры с *бинарными отношениями* допускают случай графического изображения
 - элементы множества изображаются точками или кружками;
 - пары (a_i, a_j) , для которых отношение истина, соединяются стрелкой от первого аргумента ко второму.



- ✓ Образ структуры с бинарными отношениями - *ориентированный граф*.

1.1. Структуры данных ...

3. Понятие структуры данных

Определение 1.3. Структуры, которым соответствует ориентированный граф с вершинами, лежащими на одной ломаной, называют *линейными*.

Есть два особых элемента:

- начальный элемент a_1 : $(\forall j) p(a_j, a_1) = \text{л}$
не имеет предшествующего элемента;
- конечный элемент a_n : $(\forall j) p(a_n, a_j) = \text{л}$
не имеет следующего элемента.

1.1. Структуры данных ...

4. Понятие схемы и экземпляра структуры данных ...

$(-8, 5, 0) \in \mathbb{R}^3$ – вектор с конкретными
числовыми значениями

$(a_1, a_2, a_3) \in \mathbb{R}^3$ – вектор – переменная

☑ *Переменная* – множество значений и имя, которому можно присвоить конкретное значение.

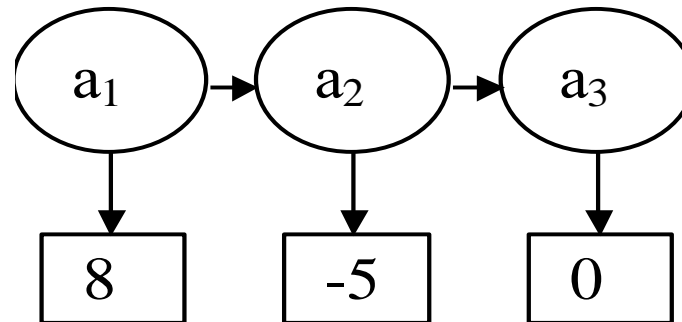
S_a – схема структуры

S_a^* – экземпляр структуры (экземпляр) в
соответствии с КОДАСИЛ

1.1. Структуры данных ...

4. Понятие схемы и экземпляра структуры данных ...

Экземпляр



Наличие конкретных значений для элементов можно выразить при помощи отношения "иметь значение"

$p_1^*(a_i, \alpha_i) = \text{и}$, если a_i имеет значение α_i

1.1. Структуры данных ...

4. Понятие схемы и экземпляра структуры данных ...

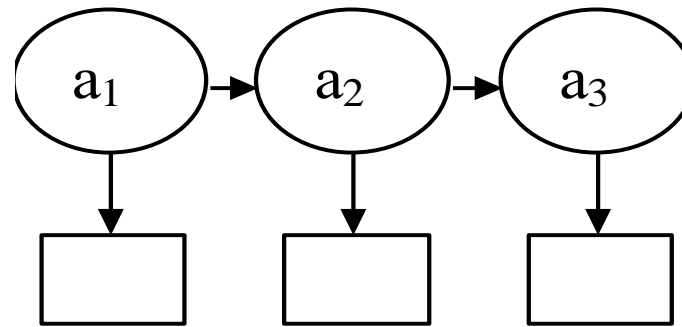
Определение 1.4. Структура данных $S_a^* = (M_a, R; p_a, p_1^*)$ с установленными значениями элементов, называется *экземпляром*.

- ☑ p_1^* - отражает не отношения между элементами, а индивидуальные свойства элемента
- ☑ Изолированные элементы множества, не связанные стрелками, не изображаются на графе.

1.1. Структуры данных ...

4. Понятие схемы и экземпляра структуры данных ...

Схема структуры



Отношение p_1 является переменной величиной.

Определение 1.5. Структура данных $S_a = (M_a, R; p_a, p_1)$, которая соответствует рассмотрению структуры как переменной величины, называется *схемой структуры*.

1.1. Структуры данных ...

4. Понятие схемы и экземпляра структуры данных

Определение 1.6. Отношения делят на две части:

- отношения, описывающие отношение следования элементов и необходимые для рекурсивно описанных операций, называемые *основными* (по ним и ведется классификация структур);
- прочие отношения описывают индивидуальные свойства элементов, и называются *вспомогательными*.

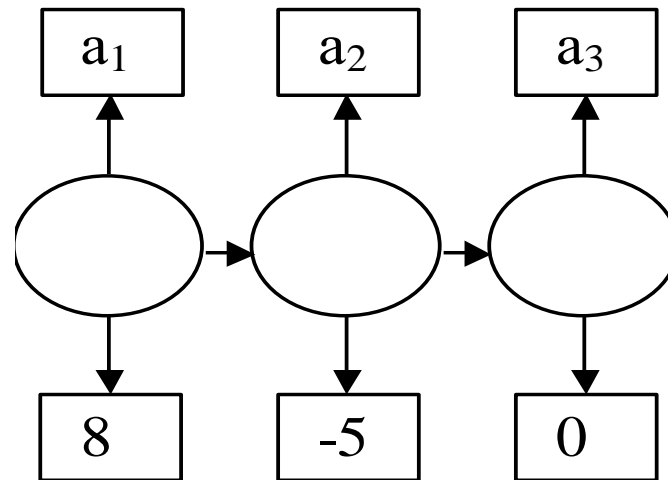
- ☑ Алгоритм соответствует схеме структуры.
Вычисления соответствует экземпляру.



1.1. Структуры данных ...

5. Именованное элементов структуры...

Различимость элементов данных, необходимая для указания этих элементов, обеспечивается путем присваивания им уникальных имен.



1.1. Структуры данных ...

5. Именованние элементов структуры...

Наличие имен для элементов можно выразить при помощи отношения "*иметь имя*".

N – множество имен

p_2 – отношение "*иметь имя*"

$S_a^* = (M_a, R, N; p_a, p_1^*, p_2^*)$ – экземпляр

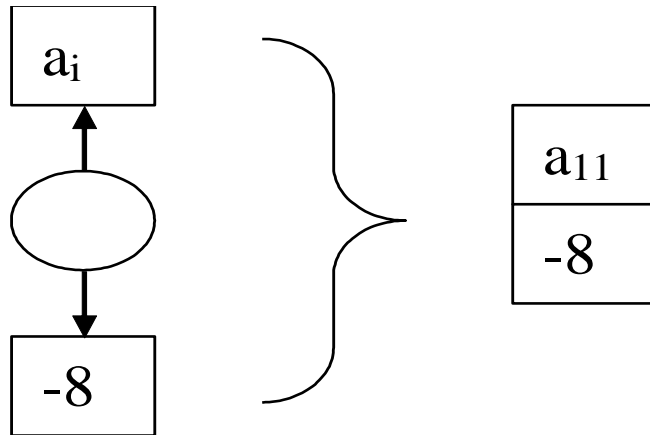
$S_a = (M_a, R, N; p_a, p_1, p_2)$ – схема

1.1. Структуры данных

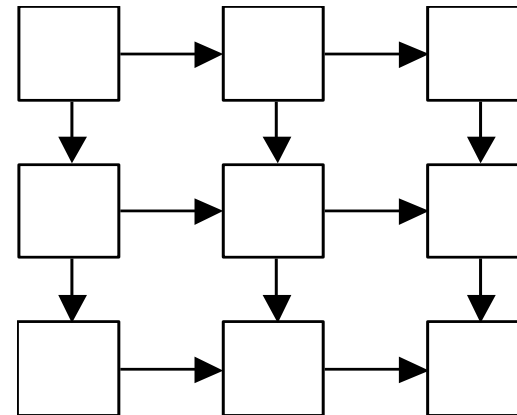
5. Именованние элементов структуры

Пример 1.2. Матрица $A=(a_{ij})$

Элемент матрицы



Матрица



⌚ Формальное определение матрицы ?

⌚ Есть ли начальные и конечные элементы ?

Заключение

- Понятие структуры данных
- Линейные структуры
- Схема и экземпляр структуры
- Базисные и вспомогательные отношения
- Примеры структур данных

Вопросы для обсуждения

- Роль выделения структур данных при разработке программ
- Способы представления структур в ЭВМ

Следующая тема

- Структуры хранения данных

Контакты

Нижегородский государственный университет
им. Н.И. Лобачевского (www.unn.ru)

Институт информационных технологий, математики
и механики (www.itmm.unn.ru)

603950, Нижний Новгород, пр. Гагарина, 23,
р.т.: (831) 462-33-56,

Гергель Виктор Павлович

(<http://www.software.unn.ru/?dir=17>)

E-mail: gergel@unn.ru

