

Основы программирования и баз данных



Модуль 1.

БАЗОВЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ

- Задача. Решение задачи
- Алгоритм. Свойства алгоритмов
- Программа. Программное обеспечение
- Информатика. Информация. Информационная технология
- Данные. Числа в арифметике
- Выражения. Операнды. Знаки операций. Идентификаторы. Константы
- Законы арифметики:
 - коммутативность
 - ассоциативность
 - дистрибутивность

Задача. Решение задачи

- **Задача**

- Поставленная цель, которую стремятся достигнуть.
- Вопрос, требующий решения на основании определённых знаний и размышления

- **Решение задачи (проблемы)**

- Процесс решения имеет место, когда ЛПР - лицо, принимающее решение (организм или система искусственного интеллекта), не знает, как перейти из данного состояния в желаемое целевое состояние.

Материал из Википедии — свободной энциклопедии

- В области вычислительной техники и программирования:
 - **Задача** - это преобразование информации (исходных данных в результирующие данные)
 - Обычно **решение задачи** представляется в виде **алгоритма**

Алгоритм

- «Алгоритм — это всякая система вычислений, выполняемых по строго определённым правилам, которая после какого-либо числа шагов заведомо приводит к решению поставленной задачи.»
(А. Колмогоров)
- «Алгоритм — это точное предписание, определяющее вычислительный процесс, идущий от варьируемых исходных данных к искомому результату.»
(А. Марков)
- «Алгоритм — это последовательность действий, направленных на получение определённого результата.»
- Алгоритм — это точный набор инструкций, описывающих порядок действий некоторого исполнителя для достижения результата — решения некоторой задачи.

Материал из Википедии — свободной энциклопедии

Свойства алгоритма

- **ПОНЯТНОСТЬ**
алгоритм для исполнителя должен включать только те команды, которые ему (исполнителю) доступны, которые входят в его систему команд.
- **определённость**
в каждый момент времени следующий шаг работы однозначно определяется состоянием системы.
- **конечность**
алгоритм должен завершать работу за конечное число шагов.
- **результативность**
при завершении работы алгоритм должен выдавать результат - решение задачи
- **массовость (исходные данные)**
алгоритм пригоден для применения к некоторому классу входных данных (из допустимого множества)
- **эффективность**
эффективный алгоритм должен потреблять минимум ресурсов (времени, памяти и т.д.)

Алгоритм и его свойства

- Ввиду важности, подчеркнем еще раз:
 - Понятие **алгоритма** — одно из основных в программировании и информатике.
 - Это последовательность команд, предназначенная исполнителю, в результате выполнения которой он должен решить поставленную задачу.
 - Алгоритм должен описываться на формальном языке, исключающем неоднозначность толкования. Исполнитель может быть человеком или машиной.
 - Исполнитель должен уметь выполнять все команды, составляющие алгоритм. Множество возможных команд конечно и изначально строго задано. Действия, выполняемые по этим командам, называются элементарными.
 - Запись алгоритма на формальном (алгоритмическом) языке называется **программой**

Материал из Википедии — свободной энциклопедии

Программа

- **Компьютерная программа** — последовательность формализованных **инструкций** для выполнения устройством управления электронно-вычислительной машины (ЭВМ, компьютера), чаще всего оформленная в виде отдельного файла (исполняемого модуля).
- Инструкции программы записываются при помощи **машинного кода** или специальных **языков программирования**.
- Процесс составления компьютерных программ носит название — **программирование**, а люди, занимающиеся этим видом деятельности именуется **программистами**.

Материал из Википедии — свободной энциклопедии

Программа (продолжение)

- Запись программы при помощи языка программирования удобна для понимания и редактирования человеком-программистом
- Для выполнения на компьютере программу необходимо **транслировать** - перевести в машинный код.
- Существует два метода трансляции:
 - **компиляция**
 - готовая программа вначале переводится в исполняемый машинный код специальной программой - **компилятором**
 - затем машинный код может многократно выполняться процессором
 - *Заметим, что машинный (двоичный) код непереносим между различными аппаратными и программными платформами.*
 - **интерпретация.**
 - программа переводится в инструкции машинного кода инструкция за инструкцией
 - каждая инструкция сразу выполняется специальной средой исполнения - **интерпретатором.**
 - *Это позволяет добиться переносимости программ между различными аппаратными и программными платформами.*
- Программы, интерпретацию которых выполняет операционная система компьютера или другие специальные программы, называются **скриптами** или **сценариями.**

Программа (продолжение)

- Большинство компьютерных программ состоят из списка инструкций, точно описывающих заложенный алгоритм
 - такой подход именуется **императивным программированием**
 - сами программы называются **императивными программами**
- Альтернативным вариантом является описание в программе характеристик исходных данных и требуемых (целевых) результатов и предоставление выбора исполняемого алгоритма компьютеру или другой выполняемой программе
 - такой подход именуется **декларативным программированием**
 - сами программы называются **декларативными программами.**

Материал из Википедии — свободной энциклопедии

Программное обеспечение

- **Программное обеспечение** — важнейшая составляющая информационных технологий, предназначенная для решения определённого круга задач.
 - В компьютерном жаргоне часто используется слово «софт» от английского **software**, которое, предположительно, в этом смысле впервые применил Джон Тьюки (John W. Tukey) в 1957 г.
- В области вычислительной техники и программирования **программное обеспечение** — это совокупность всей информации: данных и программ, - которые обрабатываются компьютерными системами, а также технической документации (руководств) по эксплуатации этих программ.
- В зависимости от назначения выделяют три **категории** программного обеспечения.

Материал из Википедии — свободной энциклопедии

Программное обеспечение (продолжение)

- **Прикладное программное обеспечение (прикладное ПО)** — это программы, предназначенные для выполнения определенных пользовательских задач и рассчитанные на непосредственное взаимодействие с пользователем.
- Пользователи компьютеров используют такие программы для выполнения конкретных **прикладных задач**:
 - подготовка и оформление документов,
 - математические вычисления,
 - обработка изображений,
 - автоматизированное проектирование,
 - бухгалтерский учет
 - и т.п.

Соответствующие программные средства называют **прикладными программами**.

Материал из Википедии — свободной энциклопедии

Программное обеспечение (продолжение)

- В отличие от *прикладного* программного обеспечения, **системное** программное обеспечение берёт на себя поддержку базовой функциональности компьютера и обеспечения работы прикладных программ.
- **Системное программное обеспечение** — это набор программ, которые:
 - управляют компонентами **вычислительной системы**, такими как процессор, коммуникационные и периферийные устройства,
 - предназначены для обеспечения функционирования и работоспособности всей системы.
 - к системному программному обеспечению относят:
 - операционные системы,
 - драйверы устройств,
 - загрузчики,
 - инструментальные программные средства (напр., DDK)
 - и др. утилиты.

Программное обеспечение (продолжение)

- **Инструментальное программное обеспечение** — программное обеспечение, предназначенное для использования в ходе проектирования, разработки и сопровождения программ:
 - Текстовые редакторы,
 - Компиляторы, интерпретаторы, ассемблеры
 - Линковщики
 - Отладчики
 - Интегрированные среды разработки
 - SDK
 - Профилировщики
 - Средства автоматизации тестирования
 - Системы управления версиями
 - и др.

Обычно термин *инструментальное* применяется для акцентирования отличия данного класса ПО от *прикладного* и *системного* программного обеспечения

Информатика

- Информатика не более наука о компьютерах, чем астрономия — наука о телескопах.
Эдсгер Дейкстра
- В широком смысле **информатика** есть наука о вычислениях, хранении и обработке **информации**.
- Она включает дисциплины, так или иначе относящиеся к вычислительным машинам: как абстрактные, вроде **анализа алгоритмов**, так и довольно конкретные, например, **разработка языков программирования**
- Темами исследований в информатике являются вопросы:
 - что можно, а что нельзя реализовать в программах (**теория вычислимости** и **искусственный интеллект**),
 - каким образом можно решать специфические задачи с максимальной эффективностью (**алгоритмы**),
 - в каком виде следует хранить и восстанавливать информацию специфического вида (**структуры данных**),
 - как программы и люди должны взаимодействовать друг с другом (**пользовательский интерфейс** и **языки программирования**)
 - и т.п.

Материал из Википедии — свободной энциклопедии

Информатика (продолжение)

- Теоретический минимум по информатике в соответствии с международными программами Computing Curricula 2001 и Software Engineering 2004, а также российскими образовательными стандартами включает следующие разделы:
 - **AL. Алгоритмы и теория сложности**

Важной составной частью знаний в области информатики является способность выбирать алгоритм, подходящий для решения данной задачи, или доказать, что такого алгоритма не существует.

Эффективность является важнейшим вопросом в данной области.
 - **AR. Архитектура и организация ЭВМ**

Понимание архитектуры и организации компьютера позволяет писать более эффективные программы.
 - **CN. Вычислительная математика и численные методы**

Научные вычисления утвердились в качестве самостоятельной дисциплины, имеющей тесные связи с информатикой.
 - **DS. Дискретные структуры**

Данный раздел включает материал из теории множеств, логики, теории графов, комбинаторики и представляет важные классы алгоритмов для решения задач в этих областях.

Материал из Википедии — свободной энциклопедии

Информатика (продолжение)

- **GV. Компьютерная графика и визуализация**

Раздел состоит из следующих взаимосвязанных частей:

- **Компьютерная графика** — вопросы генерации, представления и использования изображений с помощью компьютеров.
- **Визуализация** — выявление и представление структур и отношений в наборах данных, относящихся к определенной области (информатике, медицине и т.п.).
- **Виртуальная реальность** — создание трехмерной среды для обеспечения лучшего взаимодействия между человеком и компьютерным миром.

- **НС. Взаимодействие человека и машины**

Основной упор в данном разделе делается на понимании взаимодействия человека с интерактивными объектами, а также на вопросах проектирования **человеко-машинных интерфейсов** для различных видов программного обеспечения.

- **IM. Управление информацией**

Этот раздел включает в себя такие вопросы, как:

- сбор, организация, преобразование и выдача информации,
- алгоритмы для эффективного доступа к хранимой информации,
- методы физического хранения информации,
- вопросы безопасности, конфиденциальности, целостности и защиты данных.

Материал из Википедии — свободной энциклопедии

Информатика (продолжение)

● IS. Интеллектуальные системы

Интеллектуальная система должна действовать рационально в изменяющейся внешней среде с целью выполнения поставленных перед ней задач, а также взаимодействовать с другими системами и с человеком.

Исследования в области **искусственного интеллекта** сосредоточены на решении таких задач, как:

- эвристический поиск и алгоритмы планирования,
- формализация способов представления знаний и рассуждений
- проблемы распознавания речи и понимания естественного языка,
- компьютерное зрение, робототехника.

● NS. Распределенные вычисления

Достижения в области сетей и телекоммуникаций увеличили значение **сетевых технологий** в компьютерных приложениях:

- методы и протоколы компьютерных коммуникаций,
- мультимедиа-системы,
- стандарты и технологии Web,
- сетевая безопасность,
- беспроводные и мобильные компьютеры
- распределенные системы.

Материал из Википедии — свободной энциклопедии

Информатика (продолжение)

- **OS. Операционные системы**

Знание принципов проектирования современных операционных систем необходимо для понимания возможностей и эффективного использования той среды, в которой выполняются программы.

- **PF. Основы программирования**

Данный раздел включает в себя материал по **фундаментальным концепциям** программирования, основным структурам данных и алгоритмам — знаний, которые важны для практики программирования независимо от применяемой парадигмы и языка программирования.

- **PL. Языки программирования**

Языки программирования являются основным средством общения программиста и компьютера. Программисты должны не просто уметь написать программу на каком-либо одном языке, они должны понимать **различные стили** программирования, присущие разным языкам.

На протяжении своей профессиональной карьеры программист использует множество различных языков и стилей. Понимание разнообразия языков программирования и различных **парадигм** значительно облегчает быстрое освоение новых языков.

Материал из Википедии — свободной энциклопедии

Информатика (продолжение)

- **SE. Программная инженерия**

Программная инженерия — это дисциплина, имеющая целью эффективное **построение программных систем**, удовлетворяющих требованиям пользователей и клиентов.

Данная дисциплина изучает все **фазы жизненного цикла** программной системы:

- анализ требований,
- разработка спецификаций,
- проектирование,
- построение,
- тестирование,
- эксплуатация и сопровождение.

- **SP. Социальные и профессиональные вопросы**

- история информатики,
- социальный контекст информатики,
- риски, связанные с их применением компьютерных систем,
- интеллектуальная собственность,
- компьютерные преступления,
- экономические вопросы, связанные с применением компьютеров.

Материал из Википедии — свободной энциклопедии

Информация

- **Информация** (от лат. **informatio** — сведение, осведомленность) — это **сведения (данные)**, которые воспринимаются живым существом или устройством и **сообщаются** (получаются, передаются, преобразуются, кодируются, регистрируются) с помощью **знаков** символьного, иконического, жестового или звукового типа.
- Информация по-разному трактуется и изучается в различных областях знания.
- В теории управления (кибернетике) и теории информации в качестве информации рассматриваются те сообщения, которые система получает из внешнего мира в процессе адаптивного управления (приспособления). Информация уменьшает общую **неопределенность** и измеряется как **мера устранения неопределенности** в системе.

Материал из Википедии — свободной энциклопедии

- Иначе **информация** - это сведения (данные), содержащиеся (закодированные) в сигналах (сообщениях).
- В программировании **программа извлекает** информацию, содержащуюся в **исходных данных**, и **преобразует** ее в форму **конечного результата**.

Информационная технология

- **Информационные технологии (ИТ, от англ. *information technology*, ИТ)** — широкий класс дисциплин и областей деятельности, относящихся к технологиям управления и обработки данных вычислительной техникой.
- Обычно под информационными технологиями понимают **компьютерные технологии**. В частности, ИТ имеют дело с использованием компьютеров и программного обеспечения.
- Специалистов по компьютерной технике и программированию часто называют ИТ-специалистами.

Материал из Википедии — свободной энциклопедии

- **Информационная технология** - система методов и способов сбора, накопления, хранения, преобразования, защиты, поиска, обработки и представления информации (*данных*).

Данные

- **Данные** (от лат. *data*) — это представление фактов и идей в формализованном виде, пригодном для передачи и обработки в некотором **информационном процессе**.
- С точки зрения программиста данные — это часть программы, совокупность значений определенных ячеек памяти, преобразование которых осуществляет программный код.
- С точки зрения компилятора, процессора, операционной системы данные — это совокупность ячеек памяти, обладающих определёнными свойствами (возможность чтения и/или записи, невозможность исполнения).
- В соответствии с **принципом фон Неймана**, одна и та же область памяти может выступать как в качестве данных, так и в качестве исполнимого кода.

Данные (продолжение)

- Традиционно выделяют два типа данных —
 - **двоичные** (бинарные),
 - **текстовые.**
- **Двоичные данные**
 - обрабатываются только специализированным программным обеспечением, знающим их структуру,
 - все остальные программы передают данные без изменений.
- **Текстовые данные**
 - воспринимаются передающими системами как текст, записанный на каком-либо языке. Для них может осуществляться:
 - **перекодировка** (из кодировки отправляющей системы в кодировку принимающей),
 - **форматирование** (замена символов переноса строки, изменение максимальной длины строки, количества пробелов в тексте)
 - и т.п.

Данные (продолжение)

- Текстовые данные, в отличие от двоичных, легко *переносятся между различными платформами*, поэтому специальный текстовый формат - **XML** - принят в качестве стандарта для обмена информацией в распределенных Интернет-приложениях.
- **XML** (*eXtensible Markup Language* — расширяемый язык разметки):
 - рекомендован Консорциумом Всемирной паутины (**W3C**) для:
 - хранения структурированных данных
 - для обеспечения совместимости при передаче структурированных данных между разными системами обработки информации, особенно при передаче таких данных через Интернет.

Числа в арифметике

- **Число** — это абстрактная сущность, используемая для описания количества.
- **Арифметика** — элементарный раздел математики, изучающий простейшие виды **чисел** (натуральные, целые, рациональные) и простейшие арифметические **операции** над ними (сложение, вычитание, умножение, деление).
- **Числа:**
 - **Натуральные** числа $1, 2, \dots$ используются для **счёта** объектов. Множество натуральных чисел обозначается N .
 - **Целые.** Если к *натуральным* числам добавить ещё **отрицательные** числа и **ноль**, мы получим множество целых чисел Z . Целые числа в математике изучаются в рамках *теории чисел*.
 - В *элементарной теории чисел* изучаются такие вопросы, как:
 - делимость целых чисел,
 - алгоритм Евклида вычисления наибольшего общего делителя,
 - разложение числа на *простые множители*,
 - совершенные числа
 - и др.

Числа в арифметике (продолжение)

- Отношения *целых* чисел называются **рациональными числами**, или обыкновенными дробями. Множество всех рациональных чисел обозначается Q .
- Следующие обобщения лежат уже за пределами арифметики:
 - **Вещественные.** Если к *рациональным* числам добавить все бесконечные и непериодические десятичные дроби, называемые *иррациональными* числами, мы получим **вещественные** числа R .
 - **Комплексные.** *Действительные* числа, в свою очередь, путем добавления квадратного корня из **-1** могут быть расширены до **комплексных чисел** C .

Материал из Википедии — свободной энциклопедии

- В программировании встречаются все (или почти все) перечисленные виды чисел, в языках программирования они представляются различными **типами** данных.

Числа в арифметике (продолжение)

- **Операции над числами:**

- **Сложение** — одна из основных операций в разных разделах математики, позволяющая объединить два объекта (в арифметике — два числа), при которой двум аргументам (*слагаемым*) a и b сопоставляется результат (*сумма*), обычно обозначаемая $a + b$.
- **Вычитание** — операция, обратная сложению. Обозначается знаком минус « $-$ ».
 - В выражении $x - y$ элемент x называется *уменьшаемым*, y - *вычитаемым*, а результат вычитания называется *разностью* x и y .
 - В области положительных чисел вычитание не всегда выполнимо (из меньшего числа нельзя вычесть большее). Это обстоятельство является формальным поводом для введения в арифметику нуля и отрицательных чисел.
 - Вычитание удобно рассматривать как разновидность сложения положительного и отрицательного чисел.
К примеру, $7 - 3$ тождественно $7 + (-3)$.

Материал из Википедии — свободной энциклопедии

Числа в арифметике (продолжение)

- **Операции над числами:**
 - **Умножение.** В арифметике под умножением понимают краткую запись суммы одинаковых слагаемых.
 - Например, запись $5*3$ обозначает «5 сложить с собой 3 раза», то есть является просто краткой записью для $5+5+5$.
 - Результат умножения называется *произведением*, а множители — *множителями* или *сомножителями*.
 - **Деление** — это действие, обратное умножению и заменяет неоднократно повторенное вычитание, например:
 - сколько раз 3 содержится в 14? Повторяя вычитание, находим, что 3 «входит» в 14 четыре раза, и еще «остается» число 2.
 - При *целочисленном* делении число 14 называется *делимым*, число 3 — *делителем*, число 4 — *частным* и число 2 — *остатком*.
 - По правилам арифметики деление на 0 *запрещено*.
 - Целое число, на которое делятся без остатка два или несколько чисел, называется их *общим делителем*.
 - В области рациональных чисел результатом деления одного целого числа на другое всегда является рациональное число.

Материал из Википедии — свободной энциклопедии

Выражения. Операнды.

- **Выражение** - математическая формула или иная символическая запись, содержащая информацию о способе вычисления искомого значения.
- Синтаксически выражение строится из *операндов* и *операторов* (знаков операций).
- **Операнд** в языках программирования — аргумент операции, т.е. значение, участвующее в вычислении.
 - В зависимости от положения операнда относительно знака операции операции подразделяются на:
 - префиксные, например, $-x$,
 - инфиксные, например, $a + b$,
 - постфиксные, например, x^3 .
 - В зависимости от числа операндов операции подразделяются на:
 - одноместные (унарные),
 - двуместные (бинарные),
 - многоместные операции.

Знаки операций.

- **Знаки операций.** В большинстве языков программирования в качестве операторов используются знаки
 - арифметических операций:
 - + сложение $a + b$
 - вычитание $a - b$
 - * умножение $a * b$
 - / деление a / b
 - изменение знака $-a$
 - операции сравнения:
 - > больше $a > b$
 - >= больше или равно $a \geq b$
 - < меньше $a < b$
 - <= меньше или равно $a \leq b$
 - = или == равно $a == b$
 - <> или != не равно $a != b$
 - другие символы (в зависимости от языка программирования)

Идентификаторы. Константы

- В качестве операндов в **выражениях** используются *идентификаторы, константы и другие выражения* (возможно, заключенные в скобки)
- **Идентификатор** (*символическое имя*)
 - это лексема (последовательность допустимых символов языка программирования, имеющая в нем смысл)
 - используется для именованности программных сущностей (*переменных, массивов, функций и др.*)
 - делает возможным ссылки на них в тексте программы
- **Константа** (постоянная величина) — некоторая величина, не изменяющая своего значения в рамках рассматриваемого процесса.
 - *Численные литералы* (например, 0, -1 или 3.14159) всегда являются константами.
- Вычисление выражений выполняется в соответствии с **приоритетами** и **ассоциативностью** операторов (операций)

Законы арифметики

- КОММУТАТИВНОСТЬ
 - **Коммутативная операция** (лат. *commutativus* — «меняющийся») — это бинарная операция, обладающая *переместительностью*: для любых элементов a и b

$$a + b = b + a$$

$$a * b = b * a$$

- АССОЦИАТИВНОСТЬ
 - **Ассоциативная операция** (лат. *associatio* — *соединение*) — это бинарная операция, обладающая *сочетательностью*: для любых элементов a , b и c

$$(a + b) + c = a + (b + c)$$

$$(a * b) * c = a * (b * c)$$

- Для ассоциативной операции результат вычисления не зависит от порядка вычисления (расстановки скобок), и потому скобки по возможности опускаются
- ДИСТРИБУТИВНОСТЬ
 - **Дистрибутивность** (лат. *distributivus* — «распределительный») — свойство *согласованности* двух бинарных операций: для любых элементов a , b и c
 - $a * (b + c) = a * b + a * c$ — дистрибутивность слева;
 - $(b + c) * a = b * a + c * a$ — дистрибутивность справа.
 - Если операция $*$ является коммутативной, как это имеет место для операции умножения, то свойства дистрибутивности слева и справа совпадают.