

П. Е. Антонюк

ИНФОРМАТИКА

Учебно-методическое пособие

**Санкт-Петербург
2024**

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«Санкт-Петербургский государственный университет
промышленных технологий и дизайна»
Высшая школа технологии и энергетики**

П. Е. Антонюк

ИНФОРМАТИКА

Учебно-методическое пособие

Утверждено Редакционно-издательским советом ВШТЭ СПбГУПТД

Санкт-Петербург
2024

УДК 004.001.025.4

ББК 60.844

A724

Рецензенты:

кандидат технических наук, доцент кафедры прикладной математики и информатики
Высшей школы технологии и энергетики Санкт-Петербургского государственного
университета промышленных технологий и дизайна,

В. П. Яковлев;

кандидат технических наук, доцент кафедры информационно-измерительных систем
и технологий СПбГЭТУ «ЛЭТИ»

Н. В. Орлова

Антонюк, П. Е.

**A724 Информатика: учебно-методическое пособие / П. Е. Антонюк. — СПб.:
ВШТЭ СПбГУПТД, 2024. — 57 с.**

Учебно-методическое пособие соответствует программе и учебному плану дисциплины «Информатика» для бакалавров очного отделения, обучающихся по направлению подготовки 01.03.02 «Прикладная математика и информатика».

Изложены вопросы, охватывающие базовые направления информатики – аппаратные и программные основы, правила информационной безопасности, понятия алгоритмов, систем счисления и логических функций. После каждой теоретической темы представлена практическая работа.

Пособие может использоваться бакалаврами и магистрами любых технических специальностей в качестве источника для самостоятельной подготовки.

УДК 004.001.025.4

ББК 60.844

© ВШТЭ СПбГУПТД, 2024

© Антонюк П. Е., 2024

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	4
ТЕМА 1. ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ ИНФОРМАТИКИ.....	5
Практическая работа № 1. История и предмет информатики.....	9
ТЕМА 2. ПРИНЦИПЫ ДЖОНА ФОН НЕЙМАНА.....	10
Практическая работа № 2. Подбор оптимальной конфигурации ЭВМ... ..	16
ТЕМА 3. КЛАССИФИКАЦИЯ СОВРЕМЕННЫХ ПРОГРАММНЫХ СРЕДСТВ.....	17
Практическая работа № 3. Лицензионные и свободно распространяемые программные продукты.....	25
ТЕМА 4. ОБЩИЕ СВЕДЕНИЯ ОБ ИНФОРМАЦИОННЫХ УГРОЗАХ КОМПЬЮТЕРА.....	27
Практическая работа № 4. Анализ примеров нарушений информационной безопасности.....	33
ТЕМА 5. АЛГОРИТМИЧЕСКИЕ ЯЗЫКИ. ОБЩИЕ СВЕДЕНИЯ О ЯЗЫКЕ ПАСКАЛЬ.....	35
Практическая работа № 5. Язык программирования Паскаль, основные операторы.....	40
ТЕМА 6. АРИФМЕТИЧЕСКИЕ ОСНОВЫ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ.....	41
Практическая работа № 6. Перевод чисел в различные системы счисления.....	44
ТЕМА 7. ЛОГИЧЕСКИЕ ОСНОВЫ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ... ..	47
Практическая работа № 7. Логические основы информатики.....	55
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	57

ВВЕДЕНИЕ

Современный мир немыслим без гигантских объемов информации, которая обрабатывается и применяется во всех сферах жизни человека. Технологии работы с информационными потоками развиваются стремительно и в различных отраслях.

Для работы с вычислительной техникой человеку необходимы знания об аппаратной основе компьютера, используемых логических и математических процедурах и функциях, программных средствах, а также о возможных угрозах безопасности. Все это изучает базовая дисциплина любого информационного направления – «Информатика».

Цель изучения дисциплины «Информатика» – приобретение студентами знаний в области методов формирования информационных потоков и технологий для использования в различных отраслях знаний.

Задачи дисциплины – научить студентов:

- работать с многофункциональными программными комплексами для управления предприятием;
- использовать базовые понятия информатики в профессиональной деятельности;
- принципам применения современных информационных технологий при организации вычислительных систем.

Настоящее учебно-методическое пособие содержит лекции, позволяющие изучить всю основную тематику информатики. После каждой лекции приведен текст практической работы, выполняемой по итогам прохождения соответствующей темы.

ТЕМА 1. ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ ИНФОРМАТИКИ

Информация (лат. informatio – разъяснение, изложение) – одно из наиболее общих понятий науки, обозначающее некоторые сведения, совокупность каких-либо данных, знаний и тому подобное. Данное определение является общенаучным, связанным с любой отраслью человеческого знания. Сузим его для применения в сфере, связанной с вычислительной техникой.

Информация – любые сведения, являющиеся объектом хранения, обработки и передачи.

На практике информация всегда представляется в виде сообщения. Информационное сообщение связано с источником сообщения, получателем сообщения и каналом связи.



Рисунок 1 – Общая схема передачи информации

Сообщение от источника к приемнику передается в виде сигналов (электрический, звуковой, световой и т. д.). Человек воспринимает сигналы органами чувств. Приемники информации в технике воспринимают сообщения с помощью различной измерительной и регистрирующей аппаратуры. В обоих случаях с приемом информации связано изменение во времени некоторой величины, характеризующей состояние приемника. Поэтому информационное сообщение может быть представлено функцией $z(t)$, характеризующей изменение во времени параметров физической среды, в которой осуществляются информационные процессы. Если эта вещественная функция $z(t)$ непрерывна, то имеет место непрерывная или *аналоговая* информация (примеры: постоянное измерение температуры, давления, влажности воздуха и т. п.). Если функция $z(t)$ не является непрерывной, то информационные сообщения, используемые человеком, имеют характер *дискретных* сообщений (примеры: сигналы тревоги, сообщения, передаваемые с помощью жестов, и т. п.).

Информация обычно обрабатывается на вычислительных машинах. Дадим определение. *Компьютер (вычислительная машина)* – устройство преобразования информации посредством выполнения управляемой программой последовательности операций.

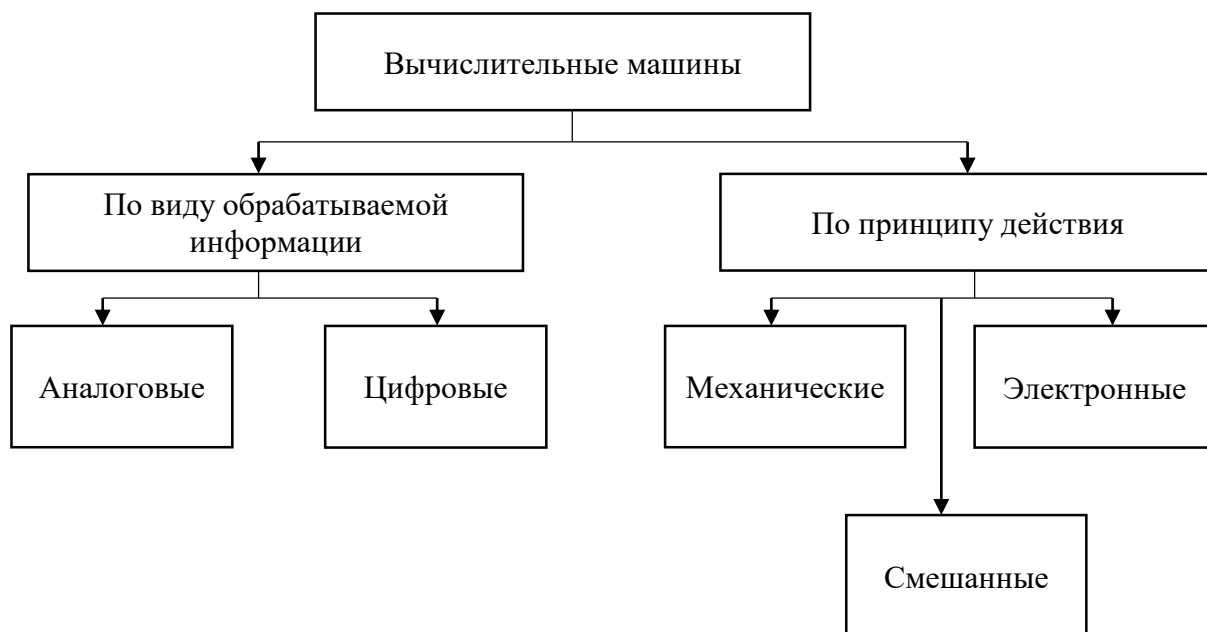


Рисунок 2 – Классификация вычислительных машин

Аналоговая вычислительная машина (АВМ) – машина, оперирующая информацией, представленной в виде непрерывных изменений некоторых физических величин. В качестве подобных переменных могут использоваться сила тока, угол поворота механического вала, скорость и ускорение движения тела.

Цифровая вычислительная машина (ЦВМ) – машина, оперирующая информацией, представленной в дискретном виде.

Из представленных определений видно, что АВМ служат для решения задач определенного класса, а ЦВМ являются универсальным вычислительным средством. Самый распространенный тип ЦВМ – электронная вычислительная машина (ЭВМ).

Для представления в ЭВМ дискретной информации применяется *алфавитный способ*, основанный на использовании фиксированного конечного набора символов любой природы, называемого алфавитом. Символы из алфавита – буквы. Любая конечная последовательность букв – слово. При этом не требуется, чтобы слово обязательно имело языковое смысловое значение.

Процесс преобразования информации часто требует представлять буквы одного алфавита буквами или словами другого алфавита. Такое представление называется *кодированием*. Процесс обратного преобразования информации относительно ранее выполненного кодирования называется *декодированием*.

Еще один очень важный термин, который все мы часто видим, слышим и читаем – термин «данные». Так как мы определили информацию как базовое понятие науки, очевидно, что данные – некое подмножество понятия «информация». Термин *данные* (англ. data) используется в отношении информации, представленной в виде, позволяющем хранить, передавать или обрабатывать ее с помощью технических средств.

Теперь мы можем дать определение всей науке, которую начинаем изучать с помощью данного пособия.

Информатика – комплексная дисциплина, изучающая все аспекты разработки, создания, оценки, использования систем обработки, структуру и общие свойства научной информации.

Любая научная дисциплина имеет такие понятия, как объект науки и предмет науки. Информатика, безусловно, не исключение из правил.

Предметом информатики как фундаментальной науки выступает информационный ресурс – его сущность, законы функционирования, механизмы взаимодействия с другими ресурсами общества. Еще одной фундаментальной категорией информатики является *социальная энтропия* – мера отклонения от некоторого состояния, принимаемого за эталонное. Чем выше информационный уровень функционирования системы, то есть чем ниже энтропия, тем экономнее расходует она традиционные ресурсы производства: энергию, сырье, рабочую силу и время. *Информационная работа* – воздействие наблюдения (управляющей подсистемы) на объект путем выработки и передачи сообщений, обуславливающих удержание объекта в имеющемся исходном состоянии, а также перевод его в новое состояние – достижение новой цели. *Информационная среда* – набор условий для технологической переработки и эффективного использования знаний в виде информационного ресурса. К информационной среде относятся аппаратные средства, программное обеспечение, телекоммуникации, уровень подготовки кадров, методы и формы управления, процедуры, регламенты, юридические нормы и так далее.

Что же касается второй составляющей, то *объектом информатики* выступают автоматизированные информационные системы (АИС) различного класса и назначения. В качестве признаков классификации АИС используются: область применения, охватываемая территория, организация информационных процессов, направление деятельности, назначение, структура и другие. Например, на представленном ниже рисунке можно увидеть классификацию автоматизированных информационных систем по направлению деятельности.

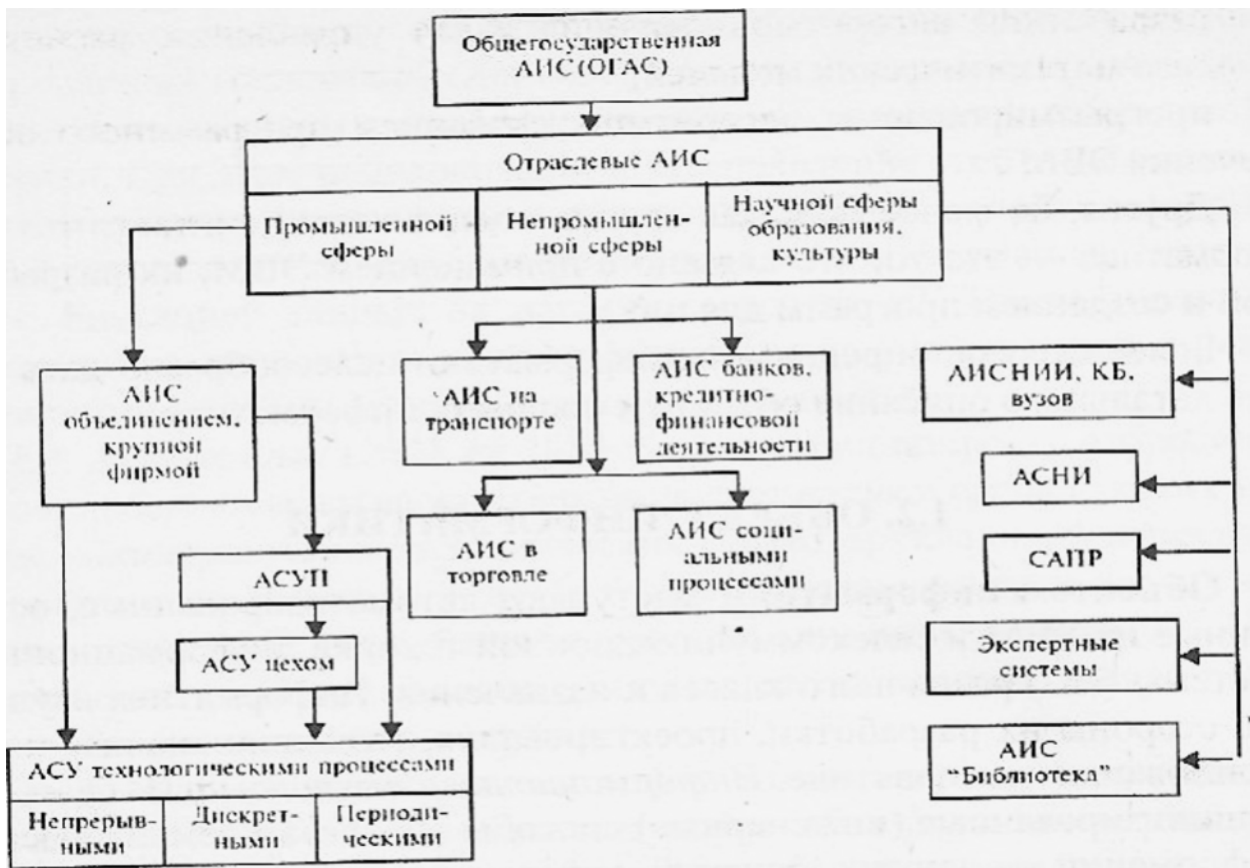


Рисунок 3 – Классификация АИС по направлению деятельности

В самом начале курса мы дали определение его базовому понятию «информация». У любого научного понятия существуют свойства, которые образуют полную группу, описывающую данное понятие. У информации имеются следующие свойства:

- 1) *релевантность* – способность информации соответствовать запросам потребителя;
- 2) *полнота* – способность информации исчерпывающе характеризовать отображаемый объект и (или) процесс;
- 3) *своевременность* – способность информации соответствовать запросам потребителя в нужный момент времени;
- 4) *достоверность* – свойство информации не иметь скрытых ошибок;
- 5) *доступность* – свойство информации, характеризующее возможность ее получения данным потребителем;
- 6) *защищенность* – свойство информации, характеризующее невозможность ее несанкционированного использования или изменения;
- 7) *эргономичность* – свойство, характеризующее удобство формы или объема информации с точки зрения данного потребителя;
- 8) *адекватность* – свойство информации однозначно соответствовать отображаемому объекту или явлениям» [1].

Практическая работа № 1. История и предмет информатики

Цель работы

Создать эссе на тему, указанную в списке согласно варианту. В представленной работе (объем не менее 7 листов формата А4) необходимо отразить суть вопроса, представить общие сведения и выразить свое мнение по данной теме.

Тематика работы

1. Информатика древних времен.
2. Информатика средних веков.
3. Информатика Нового времени.
4. Информатика современности.
5. Предметная область теоретической информатики.
6. Предметная область практической информатики.
7. Предметная область технической информатики.
8. Мировоззренческая роль информатики.
9. Воспитательная роль информатики.
10. Междисциплинарная роль информатики.
11. Общекультурная роль информатики.
12. Исторические личности в информатике.
13. Предмет информатики и его различные толкования.
14. Информатика как проявление единства и борьбы теории и практики.
15. Информатика как двигатель научно-технического прогресса.
16. Будущее информатики.

ТЕМА 2. ПРИНЦИПЫ ДЖОНА ФОН НЕЙМАНА

Большинство современных ЭВМ строится на базе принципов, сформулированных американским ученым, одним из отцов кибернетики Джоном фон Нейманом. Эти принципы сводятся к следующему:

1. Основными блоками фон-неймановской машины являются: блок управления, АЛУ (арифметико-логическое устройство), память, устройства ввода-вывода.
2. Информация кодируется в двоичной форме и разделяется на единицы, называемые словами.
3. Алгоритм представляется в форме последовательности управляющих слов, которые определяют смысл операции. Эти управляющие слова называются командами. Совокупность команд, представляющая алгоритм, называется программой.
4. Программы и данные хранятся в одной и той же памяти. Разнотипные слова различаются по способу использования, но не по способу кодирования.
5. УУ и АЛУ обычно объединяются в одно, называемое центральным процессором. Они определяют действия путем считывания команд из оперативной памяти. Обработка информации, предписанная алгоритмом, сводится к последовательному выполнению команд в порядке, однозначно определяемом программой.

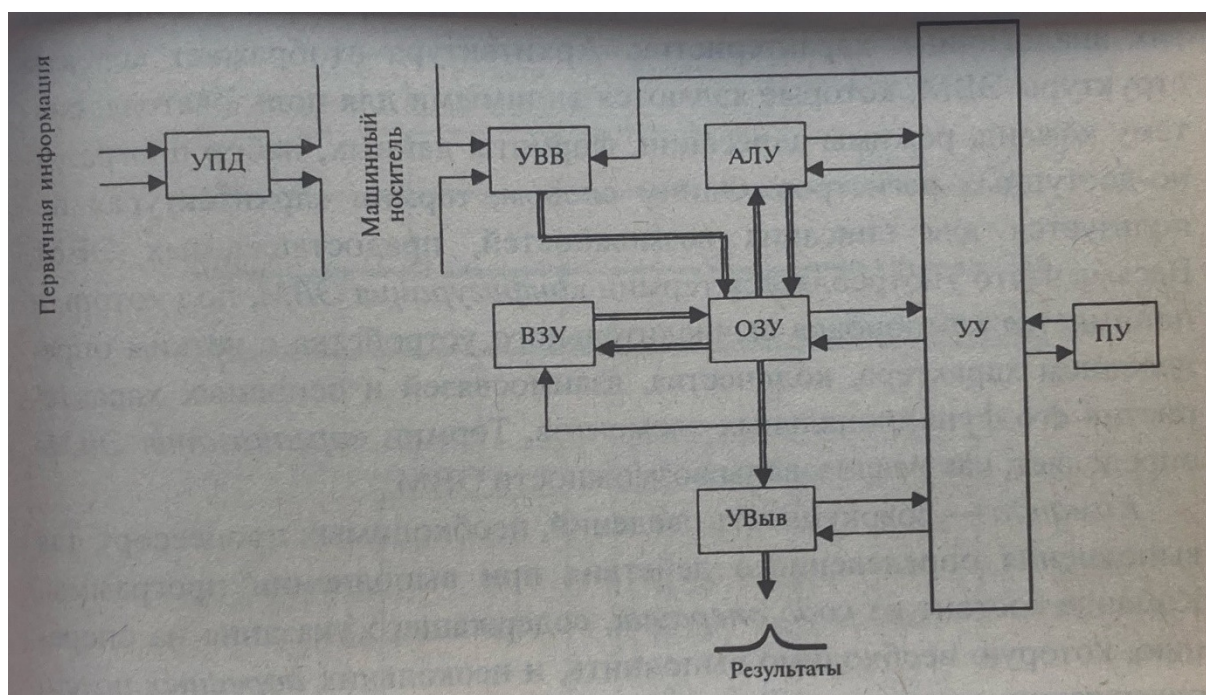


Рисунок 4 – Обобщенная структурная схема ЭВМ

Обобщенная схема организации ЭВМ представлена на рисунке 4, где УПД – устройство подготовки данных; УВВ – устройство ввода информации; УВыв – устройство вывода информации; АЛУ – арифметико-логическое

устройство; УУ – устройство управления; ПУ – пульт управления; ОЗУ – оперативное запоминающее устройство; ВЗУ – внешнее запоминающее устройство.

Архитектура ЭВМ – абстрактное определение машины в терминах основных функциональных модулей, языка, структур данных. Архитектура отображает аспекты структуры ЭВМ, видимые для пользователя: систему команд, режимы адресации, форматы данных, набор программно-доступных регистров. Команда – совокупность сведений, необходимых процессору для выполнения определенного действия при выполнении программы. Команда состоит из кода операции, содержащего указания на операцию, которую необходимо выполнить, и нескольких адресных полей, содержащих указание на места расположения операндов команды. Способ вычисления адреса по информации, содержащейся в адресном поле команды, называется режимом (типом) адресации. Множество команд, реализованных в данной ЭВМ, образует ее систему команд.

Принципы фон Неймана практически можно реализовать множеством различных способов. Большинство мини- и микро ЭВМ имеют шинную организацию, которая является простейшей формой организации ЭВМ. Упрощенную схему ЭВМ с шинной организацией можно представить следующим образом

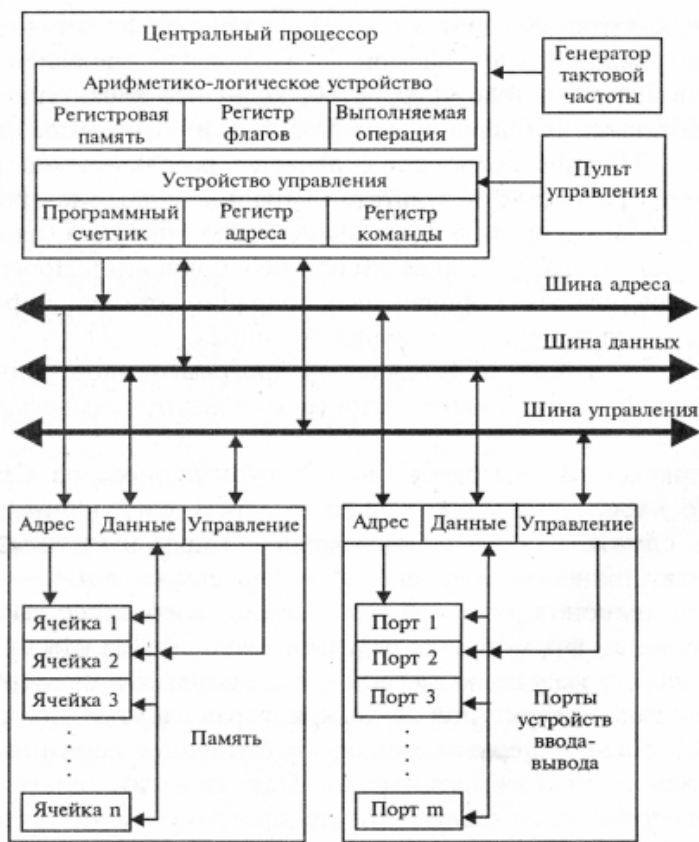


Рисунок 5 – Шинная организация (архитектура) ЭВМ

АЛУ выполняет основную работу по переработке информации, хранимой в ОЗУ. В нем выполняются арифметические и логические операции. Выполняемая функция определяется микрокомандой, получаемой от устройства управления. АЛУ содержит в своем составе устройство, хранящее характеристику результата выполнения операции над данными и называемое флаговым регистром. Отдельные разряды этого регистра указывают на равенство результата нулю, на знак результата операции (+ или -), на правильность выполнения операции (наличие переноса за пределы разрядной сетки или переполнения). Программный анализ флага позволяет производить операции ветвления программы в зависимости от конкретных значений данных. Кроме того, в АЛУ имеется набор программно-доступных быстродействующих ячеек памяти, которые называются регистрами процессора. Регистры составляют основу архитектуры процессора. Среди обязательного набора регистров можно отметить следующие. Регистр данных – служит для временного хранения промежуточных результатов при выполнении операций. Регистр-аккумулятор – регистр временного хранения, который используется в процессе вычислений (например, в нем формируется результат выполнения команды умножения). Регистр указатель стека – используется при операциях со стеком, т. е. такой структуры данных, которая работает по принципу: последним вошел – первым вышел, т. е. последнее записанное в него значение извлекается из него первым, стеки используются для организации подпрограмм. Индексные, указательные и базовые регистры используются для хранения и вычисления адресов операндов в памяти. Регистры-счетчики используются для организации циклических участков в программах. Регистры общего назначения, имеющиеся во многих ЭВМ, могут использоваться для любых целей. Точное назначение такого регистра определяет программист при написании программы. Количество регистров и связей между ними оказывает существенное влияние на сложность и стоимость процессора. Однако, с другой стороны, наличие большого количества регистров с богатым набором возможностей упрощает программирование и повышает гибкость программного обеспечения. Кроме перечисленных регистров, в состав АЛУ могут входить внутренние системные регистры, не доступные программно и используемые во время внутренних пересылок информации при выполнении команд.

Устройство управления (УУ) – часть центрального процессора, которая вырабатывает распределенную во времени и пространстве последовательность внутренних и внешних управляющих сигналов, обеспечивающих выборку и выполнение команд. На этапе цикла выборки команды УУ интерпретирует команду, выбранную из программной памяти. На этапе выполнения команды в соответствии с типом реализуемой операции УУ формирует требуемый набор команд низкого уровня для АЛУ и других устройств. Эти команды задают последовательность простейших низкоуровневых операций, таких как пересылка данных, сдвиг данных, установка и анализ признаков, запоминание результатов и др. Такие элементарные низкоуровневые операции называют микрооперациями, а команды, формируемые УУ, называются микрокомандами.

Последовательность микрокоманд, соответствующая одной команде, называется микропрограммой.

В простейшем случае УУ имеет в своем составе три устройства: регистр команды, который содержит код команды во время ее выполнения, программный счетчик, в котором содержится адрес очередной подлежащей выполнению команды, регистр адреса, в котором вычисляются адреса операндов, находящихся в памяти.

Для связи пользователя с ЭВМ предусмотрен пульт управления, который позволяет выполнять такие действия, как сброс ЭВМ в начальное состояние, просмотр регистра или ячейки памяти, запись адреса в программный счетчик, пошаговое выполнение программы при ее отладке и т.д.

В их число входят устройства двух типов: устройства внешней памяти, предназначенные для долговременного хранения данных большого объема и программ, и коммуникационные устройства, предназначенные для связи ЭВМ с внешним миром (с пользователем, другими ЭВМ и т. д.). Обмен данными с внешним устройством осуществляется через порты ввода-вывода. Порт (в переводе с англ. port – ворота, дверь, отверстие) – это абстрактное понятие, на самом деле несуществующее. По аналогии с ячейками памяти порты можно рассматривать как ячейки, через которые можно записать в периферийное устройство, или, наоборот, прочесть из него. Так же, как и ячейки памяти, порты имеют адреса портов ввода-вывода. Для обмена данными между компьютером и ПУ в компьютере предусмотрен внешний интерфейс, то есть набор проводов, соединяющих компьютер и ПУ, а также набор правил обмена информацией по этим проводам (протокол).

Объединение функциональных блоков в ЭВМ осуществляется посредством следующей системы шин: шины данных, по которой осуществляется обмен информацией между блоками ЭВМ, шины адреса, используемой для передачи адресов (номеров ячеек памяти или портов ввода-вывода, к которым производится обращение), и шины управления для передачи управляющих сигналов. Совокупность этих трех шин называют системной шиной, системной магистралью или системным интерфейсом. Состав и назначение шин и правило их использования, виды передаваемых по шине сигналов и другие характеристики шины могут существенно различаться у разных видов ЭВМ. Шина состоит из отдельных проводников (линий). Сигналы по линиям шины могут передаваться либо импульсами (наличие импульса соответствует логической 1, а отсутствие импульса – 0, либо уровнем напряжения: высокий уровень – логическая 1, низкий – 0). Шириной шины называется количество линий (проводников), входящих в состав шины. Ширина шины адреса определяет размер адресного пространства ЭВМ. Если, например, количество линий адреса, используемых для адресации памяти, равно 20, то общее количество адресуемых ячеек памяти составит 2²⁰, т. е. примерно один миллион ячеек.

Функционирование ЭВМ с шинной структурой можно описать следующим обобщенным алгоритмом.



Рисунок 6 – Алгоритм функционирования ЭВМ с шинной архитектурой

1. Инициализация. После включения ЭВМ или операции сброса в регистры центрального процессора заносятся некоторые начальные значения. Программному счетчику присваивается начальное значение, равное адресу первой команды программы, указанной выше.
2. ЦП производит операцию считывания команды из памяти. В качестве адреса ячейки памяти используется содержимое программного счетчика.
3. Содержимое считанной ячейки памяти интерпретируется процессором как команда и помещается в регистр команды. УУ приступает к интерпретации прочитанной команды. По полю кода операции из первого слова команды УУ определяет ее длину и, если это необходимо, организует дополнительные операции считывания, пока вся команда полностью не будет прочитана процессором. Вычисленная длина команды прибавляется к исходному содержимому программного счетчика, и когда команда полностью прочитана, программный счетчик будет хранить адрес следующей команды.
4. По адресным полям команды УУ определяет, имеет ли команда операнды в памяти. Если это так, то на основе указанных в адресных полях режимов адресации вычисляются адреса операндов и производится операция чтения памяти для считывания операндов.
5. УУ и АЛУ выполняют операцию, указанную в поле кода операции команды. Во флаговом регистре процессора запоминаются признаки

результата операции (равно 0 или нет, знак результата, наличие переполнения и т. д.).

6. Если это необходимо, УУ выполняет операцию записи для того, чтобы поместить результат выполнения команды в память.
7. Если последняя команда не была командой «остановить процессор», то описанная последовательность действий повторяется, начиная с шага 1. Описанная последовательность действий ЦП с шага 1 до шага 6 называется циклом процессора.

Таким образом, функционирование любой фон-неймановской ЭВМ описывается алгоритмом, близким к приведенному выше, и представляет собой последовательность достаточно простых действий, выполняемых с очень большой скоростью» [1].

Практическая работа № 2. Подбор оптимальной конфигурации ЭВМ

Цель работы

Получение навыков по выбору комплектующих устройств для своего персонального компьютера.

Последовательность выполнения работы

Создать и сохранить в личной папке файл «Практическая раб№2 Информ» следующего содержания:

- Заголовок «Отчет о выполнении практической работы №2»;
- В верхнем колонтитуле укажите фамилию и имя, а в нижнем – Практическая работа №2;
- С помощью сети Интернет подобрать конфигурацию аппаратных частей ПК для работы с офисными приложениями и заполнить таблицу:

Конфигурация ПК для работы с офисными приложениями

Название устройства	Модель устройства	Характеристика

Аналогично составить еще 4 таблицы:

- Конфигурация ПК для работы с графическими программами
- Конфигурация ПК для обработки аудиоинформации
- Конфигурация ПК для работы в сети Интернет
- Конфигурация ПК для современных компьютерных игр.

ТЕМА 3. КЛАССИФИКАЦИЯ СОВРЕМЕННЫХ ПРОГРАММНЫХ СРЕДСТВ

Программное обеспечение – неотъемлемая часть компьютерной системы. Оно является логическим продолжением технических средств. Сфера применения конкретного компьютера определяется созданным для него программным обеспечением. Сам по себе компьютер не обладает знаниями ни в одной области применения. Все эти знания сосредоточены в выполняемых на компьютерах программах. Программное обеспечение современных компьютеров включает миллионы программ – от игровых до научных.

Программное обеспечение делится на три категории:

- системное ПО (программы общего пользования), выполняющие различные вспомогательные функции, например, создание копий используемой информации, выдачу справочной информации о компьютере, проверку работоспособности устройств компьютера и т. д.

- прикладное ПО, обеспечивающее выполнение необходимых работ на ПК: редактирование текстовых документов, создание рисунков или картинок, обработка информационных массивов и т. д.

- инструментальное ПО (системы программирования), обеспечивающее разработку новых программ для компьютера на языке программирования.

К прикладному программному обеспечению относятся программы, написанные для пользователей или самими пользователями, для задания компьютеру конкретной работы.

Системное программное обеспечение обеспечивает и контролирует доступ к аппаратному обеспечению компьютера. Прикладное программное обеспечение взаимодействует с аппаратными компонентами через системное. Конечные пользователи в основном работают с прикладным программным обеспечением. Чтобы обеспечить аппаратную совместимость, каждый тип программного обеспечения разрабатывается для конкретной аппаратной платформы.

Системное программное обеспечение – это набор программ, которые управляют компонентами компьютера, такими как процессор, коммуникационные и периферийные устройства.

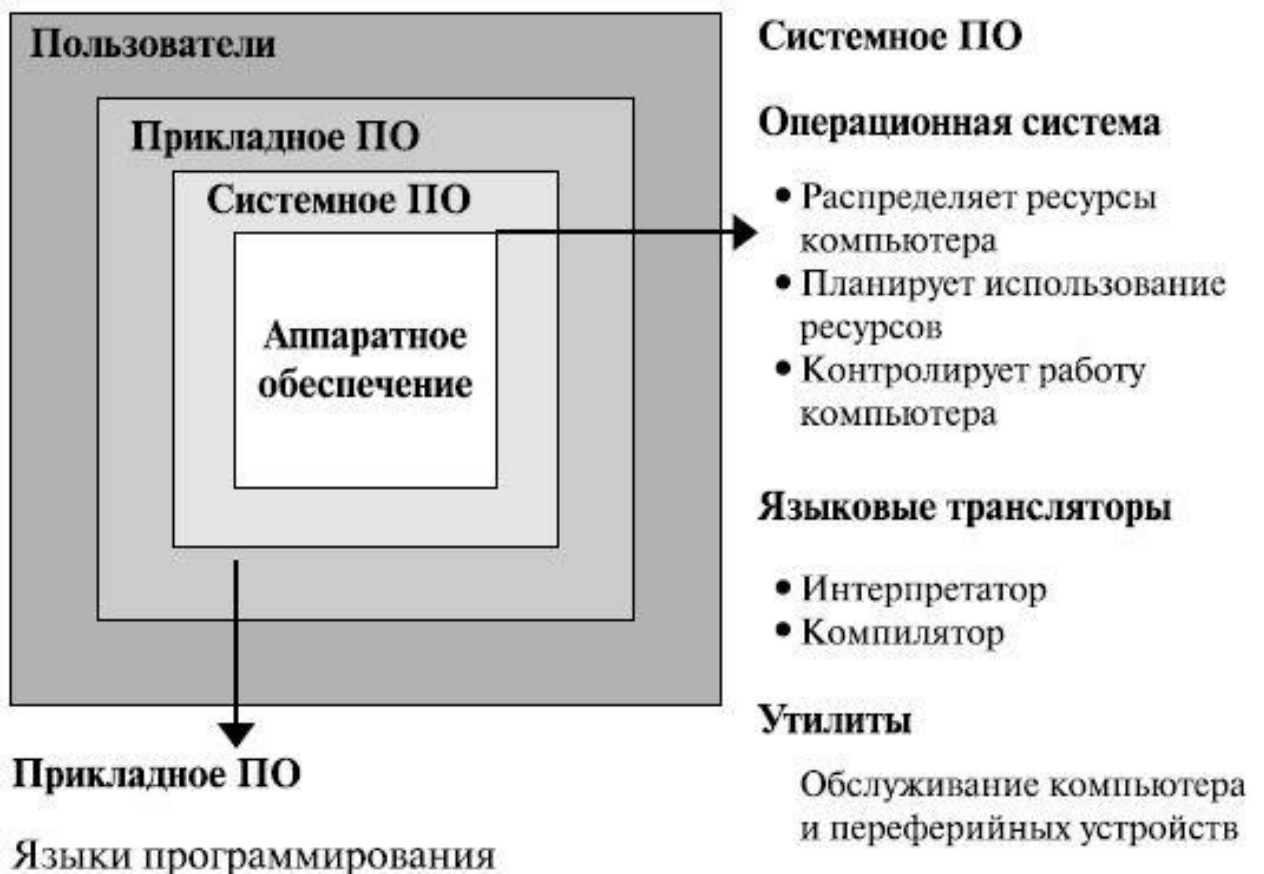


Рисунок 7 – Структура и назначение программного обеспечения

Системное программное обеспечение (System Software) – совокупность программ и программных комплексов, предназначенная для обеспечения работы компьютера и сетей ЭВМ. Системное программное обеспечение выполняет следующие задачи:

- создание операционной среды функционирования других программ;
- обеспечение надежной и эффективной работы самого компьютера и вычислительной сети;
- проведение диагностики, локализации сбоев, ошибок и отказов, и профилактики аппаратуры компьютера и вычислительных сетей;
- выполнение вспомогательных технологических процессов (копирование, архивирование);
- восстановление файлов, программ, баз данных и т. д.



Рисунок 8 – Классы программных продуктов

Системное программное обеспечение – это набор программ, которые управляют компонентами компьютера, такими как процессор, коммуникационные и периферийные устройства. Программистов, которые создают системное программное обеспечение, называют системными программистами. Эти программы общего пользования не связаны с конкретным применением ПК и выполняют традиционные функции: планирование и управление задачами, управления вводом-выводом и т. д.

Другими словами, системные программы выполняют различные вспомогательные функции, например, создание копий используемой информации, выдачу справочной информации о компьютере, проверку работоспособности устройств компьютера и т. п.

Системное программное обеспечение (рис. 9) можно разделить на базовое программное обеспечение, которое, как правило, поставляется вместе с компьютером, и сервисное программное обеспечение, которое может быть приобретено дополнительно.

Базовое обеспечение (base software) – минимальный набор программных средств, обеспечивающих работу компьютера. Сервисное программное обеспечение включает программы и программные комплексы, которые расширяют возможности базового программного обеспечения и организуют более удобную среду работы пользователя.



Рисунок 9 – Структура системного программного обеспечения

В базовое программное обеспечение входят:

- операционная система;
- операционные оболочки (обычно текстовые и графические);
- сетевая операционная система.

Операционная система предназначена для управления выполнением пользовательских программ, планирования и управления вычислительными ресурсами ЭВМ.

Сетевые операционные системы – комплекс программ, обеспечивающий обработку, передачу и хранение данных в сети. Сетевая ОС предоставляет пользователям различные виды сетевых служб (управление файлами, электронная почта, аудио- и видеоконференции, распределенные вычисления, процессы управления сетью и др.), поддерживает работу в абонентских системах. Сетевые операционные системы используют архитектуру клиент-сервер или одноранговую архитектуру. Вначале сетевые операционные системы поддерживали лишь локальные вычислительные сети (ЛВС), сейчас эти операционные системы распространяются на ассоциации локальных сетей. Операционные оболочки – специальные программы, предназначенные для облегчения общения пользователя с командами операционной системы.

Операционные оболочки имеют текстовый и графический варианты интерфейса конечного пользователя, а в будущем возможны варианты речевого интерфейса и распознавание рукописного ввода данных. Эти программы существенно упрощают задание управляющей информации для выполнения команд операционной системы, уменьшают напряженность и сложность работы конечного пользователя.

Сервисное ПО можно классифицировать по функциональному признаку следующим образом:

- программы диагностики работоспособности компьютера;
- антивирусные программы, обеспечивающие защиту компьютера, обнаружение и восстановление зараженных файлов;
- программы обслуживания дисков, обеспечивающие проверку качества поверхности магнитного диска;
- контроль сохранности файловой системы на логическом и физическом уровнях, сжатие дисков, создание страховых копий дисков, резервирование данных на внешних носителях и др.;
- программы архивирования данных, которые обеспечивают процесс сжатия информации в файлах с целью уменьшения объема памяти для ее хранения;
- программы обслуживания сети.

Эти программы также часто называются утилитами.

Утилиты – программы, служащие для выполнения вспомогательных операций обработки данных или обслуживания компьютеров (диагностики, тестирования аппаратных и программных средств, оптимизации использования дискового пространства, восстановления разрушенной на магнитном диске информации и т. п.).

Прикладные программы предназначены для того, чтобы обеспечить применение вычислительной техники в различных сферах деятельности человека. Помимо создания новых программных продуктов, разработчики прикладных программ большие усилия тратят на совершенствование и модернизацию популярных систем, создание их новых версий. Новые версии, как правило, поддерживают старые, сохраняя преемственность, и включают в себя базовый минимум (стандарт) возможностей.

К прикладному программному обеспечению относятся программы, написанные для пользователей или самими пользователями, для задания компьютеру конкретной работы. Программы обработки заказов или создания списков рассылки – примеры прикладного программного обеспечения.

К прикладному ПО, например, относятся:

- комплект офисных приложений MS OFFICE;
- бухгалтерские системы;
- редакторы HTML или Web-редакторы;
- браузеры – средства просмотра Web-страниц;
- графические редакторы.

Один из возможных вариантов классификации программных средств (ПС), составляющих прикладное программное обеспечение (ППО), отражен на рис. 10.



Рисунок 10 – Структура прикладного программного обеспечения

Несмотря на широкие возможности использования компьютеров для обработки самой разной информации, самыми популярными являются программы, предназначенные для работы с текстами – текстовые редакторы и издательские системы.

Текстовыми редакторами называют программы для ввода, обработки, хранения и печатания текстовой информации в удобном для пользователя виде. Эксперты оценивают использование компьютера в качестве печатающей машинки в 80 % всего времени задействования техники.

Большую популярность приобрели программы обработки графической информации. Компьютерная графика в настоящее время является одной из самых динамично развивающихся областей программного обеспечения. Она включает в себя ввод, обработку и вывод графической информации – чертежей, рисунков, фотографий, картин, текстов и т. д. средствами компьютерной техники. Различные типы графических систем позволяют быстро строить изображения, вводить иллюстрации с помощью сканера или видеокамеры, создавать анимационные ролики. Графические редакторы позволяют пользоваться различным инструментарием художника, стандартными библиотеками изображений, наборами стандартных шрифтов, редактированием изображений, копированием и перемещением фрагментов по страницам экрана и др.

Для выполнения расчетов и дальнейшей обработки числовой информации существуют специальные программы – электронные таблицы. В процессе деятельности любого специалиста часто требуется представить результаты работы в виде таблиц, где одна часть полей занята исходными данными, а другая – результатами вычислений и графического анализа. Характерными для них является большой объем перерабатываемой информации, необходимость многократных расчетов при изменении исходных данных. Автоматизацией подобной рутинной работы и занимаются электронные таблицы.

Одним из наиболее перспективных направлений развития вычислительной техники является создание специальных аппаратных средств для хранения гигантских массивов информационных данных и последующей нечисловой обработки их, чаще всего – поиска и сортировки. Для компьютерной обработки подобных баз данных используют системы управления базами данных (СУБД). Последние представляют собой набор средств программного обеспечения, необходимых для создания, обработки и вывода записей баз данных.

К прикладному программному обеспечению относятся также инструментальные программные средства специального назначения. В настоящее время создаются различные специальные программные системы целевого назначения, предназначенные для работы специалистов в некоторой предметной области. Такие программы называют авторскими инструментальными системами. Авторская система представляет интегрированную среду с заданной интерфейсной оболочкой, которую пользователь может наполнить информационным содержанием своей предметной области.

Среди таких систем получили распространение экспертные системы. Такие программы ведут себя подобно эксперту в некоторой узкой прикладной области. Экспертные системы призваны решать задачи с неопределенностью и неполными исходными данными, требующие для своего решения экспертных знаний. Кроме того, эти системы должны уметь объяснять свое поведение и свое решение. Экспертные системы, являющиеся основой искусственного интеллекта, получили широкое распространение в различных областях науки (например, для классификации животных и растений по видам, для химического анализа), в медицине (постановка диагноза, анализ электрокардиограмм, определение методов лечения), в технике (поиск неисправностей в технических устройствах, слежение за полетом космических кораблей и спутников), в политологии и социологии, криминалистике, лингвистике и т. д.

В последнее время широкую популярность получили программы обработки гипертекстовой информации. Гипертекст – это форма организации текстового материала не в линейной последовательности, а в форме указания возможных переходов (ссылок), связей между отдельными его фрагментами. В обычном тексте используется обычный линейный принцип размещения

информации, и доступ к нему (тексту) осуществляется последовательно. В гипертекстовых системах информация напоминает текст энциклопедии, и доступ к любому выделенному фрагменту текста осуществляется произвольно по ссылке. Организация информации в гипертекстовой форме используется при создании справочных пособий, словарей, контекстной помощи (Help) в прикладных программах.

Отдельную группу прикладного ПО составляют программные средства профессионального уровня. Каждая прикладная программа этой группы ориентируется на достаточно узкую предметную область, но проникает в нее максимально глубоко. Так функционируют АСНИ – автоматизированные системы научных исследований, каждая из которых «привязана» к определенной области науки, САПР – системы автоматизированного проектирования, каждая из которых также работает в узкой области, АСУ – автоматизированные системы управления (которых в 1960-1970-х годах были разработаны тысячи)» [2].

Практическая работа № 3

Лицензионные и свободно распространяемые программные продукты

Неотъемлемой частью компьютера является программное обеспечение, ведь компьютер работает по программе и с программами. Сам компьютер не обладает знаниями ни в одной области применения. Все эти знания сосредоточены в выполняемых на компьютерах программах. Программное обеспечение (ПО) современных компьютеров включает в себя миллионы программ – от игровых до научных.

Под программным обеспечением (Software) понимается совокупность программ, которые могут выполняться вычислительной системой.

Программы по их правовому статусу можно разделить на три большие группы: лицензионные, условно бесплатные и свободно распространяемые.

1. *Коммерческие лицензионные программы.* В соответствии с лицензионным соглашением разработчики программы гарантируют ее нормальное функционирование в определенной операционной системе и несут за это ответственность. Лицензионные программы разработчики обычно продают в коробочных дистрибутивах. В коробке находятся CD-диски, с которых производится установка программы на компьютеры пользователей, и руководство пользователей по работе с программой. Довольно часто разработчики предоставляют существенные скидки при покупке лицензий на использование программы на большом количестве компьютеров или в учебных заведениях.

2. *Свободно распространяемые программы (Freeware).* Многие производители программного обеспечения и компьютерного оборудования заинтересованы в широком бесплатном распространении программного обеспечения. К таким программным средствам можно отнести:

- новые недоработанные (бета) версии программных продуктов (это позволяет провести их широкое тестирование);
- программные продукты, являющиеся частью принципиально новых технологий (это позволяет завоевать рынок);
- дополнения к ранее выпущенным программам, исправляющие найденные ошибки или расширяющие возможности;
- драйверы к новым или улучшенные драйверы к уже существующим устройствам.

3. *Условно бесплатные программы.* Некоторые фирмы-разработчики программного обеспечения предлагают пользователям условно бесплатные программы в целях рекламы и продвижения на рынок. Пользователю предоставляется версия программы с определенным сроком действия (после истечения указанного срока действия программы прекращает работать, если за нее не была произведена оплата) или версия программы с ограниченными функциональными возможностями (в случае оплаты пользователю сообщается код, включающий все функции программы).

Задания к практической работе

Задание № 1. Найти в Интернете закон РФ «Об информации, информатизации и защите информации» и выделить определения понятий:

1. Обладатель информации.
2. Конфиденциальность информации.

Задание № 2. Найдите в законе РФ «Об информации, информатизации и защите информации» информацию об ответственности за правонарушения в сфере информации, информационных технологий и защите информации.

Задание № 3. Заполнить таблицу «Платное и бесплатное программное обеспечение».

Таблица 1– Платное и бесплатное программное обеспечение

ПО	Платные программы	Бесплатные программы
Операционные системы		
Программы для работы с офисными документами		
Программы для работы с изображениями		
Программы для работы с видео и звуком		
Программы для записи дисков		
Программы для виртуального общения		
Программы-переводчики		
Бухгалтерские программы		
Антивирусы		
Архиваторы		
Распознавание текста		

Контрольные вопросы:

1. Что такое программное обеспечение компьютера?
2. Какие программы являются условно бесплатными?
3. Какие программные средства относят к свободно распространяемым программам?
4. В чем преимущества лицензионного программного обеспечения?
5. Какие проблемы могут возникнуть при использовании нелицензионного программного продукта?

ТЕМА 4. ОБЩИЕ СВЕДЕНИЯ ОБ ИНФОРМАЦИОННЫХ УГРОЗАХ КОМПЬЮТЕРА

В связи со стремительным развитием информационных технологий и их проникновением во все сферы человеческой деятельности возросло количество преступлений, направленных против информационной безопасности.

Большой интерес со стороны кибер-преступников вызывает деятельность государственных структур и коммерческих предприятий. Целью является хищение, разглашение конфиденциальной информации, подрыв деловой репутации, нарушение работоспособности и, как следствие, доступности информационных ресурсов организации. Данные действия наносят огромный моральный и материальный ущерб.

Однако риску подвергаются не только крупные компании, но и частные пользователи. С помощью различных средств преступники получают доступ к персональным данным – номерам банковских счетов, кредитных карт, паролям, выводят систему из строя или получают полный доступ к компьютеру. В дальнейшем такой компьютер может использоваться как часть *зомби-сети* – сети зараженных компьютеров, используемых злоумышленниками для проведения атак на сервера, рассылки спама, сбора конфиденциальной информации, распространения новых вирусов и троянских программ.

Сегодня всеми признается, что информация является ценным достоянием и подлежит защите. В то же время информация должна быть доступной для определенного круга пользователей (сотрудникам, клиентам и партнерам компаний). Таким образом, встает вопрос о создании комплексной системы информационной безопасности. Подобная система должна учитывать все возможные источники угроз (человеческий, технический и стихийный факторы) и использовать весь комплекс защитных мер, таких как физические, административные и программно-технические средства защиты.

Источники угроз

В качестве источника угроз информационной безопасности может выступать человек либо группа людей, а также некие, независящие от деятельности человека, проявления. Исходя из этого все источники угроз можно разделить на три группы:

Человеческий фактор. Данная группа угроз связана с действиями человека, имеющего санкционированный или несанкционированный доступ к информации. Угрозы этой группы можно разделить на:

- внешние, к ним относятся действия кибер-преступников, хакеров, интернет-мошенников, недобросовестных партнеров, криминальных структур.

- внутренние, к ним относятся действия персонала компаний, а также пользователей домашних компьютеров. Действия данных людей могут быть как умышленными, так и случайными.

Технический фактор. Эта группа угроз связана с техническими проблемами – физическое и моральное устаревание используемого оборудования, некачественные программные и аппаратные средства обработки информации. Все это приводит к отказу оборудования и зачастую потери информации.

Стихийный фактор. Эта группа угроз включает в себя природные катаклизмы, стихийные бедствия и прочие форс-мажорные обстоятельства, не зависящие от деятельности людей.

Все три источника угроз необходимо обязательно учитывать при разработке системы защиты информационной безопасности.

Источники распространения угроз

Развитие современных компьютерных технологий и средств связи дает возможность злоумышленникам использовать различные источники распространения угроз. Рассмотрим их подробнее:

1. Интернет

Глобальная сеть Интернет уникальна тем, что не является чьей-то собственностью и не имеет территориальных границ. Это во многом способствует развитию многочисленных веб-ресурсов и обмену информацией. Сейчас любой человек может получить доступ к данным, хранящимся в интернете, или создать свой собственный веб-ресурс.

Однако эти же особенности глобальной сети предоставляют злоумышленникам возможность совершения преступлений в интернете, при этом затрудняя их обнаружение и наказание.

Злоумышленники размещают вирусы и другие вредоносные программы на веб-ресурсах, «маскируют» их под полезное и бесплатное программное обеспечение. Кроме того, скрипты, автоматически запускаемые при открытии веб-страницы, могут выполнять вредоносные действия на вашем компьютере, включая изменение системного реестра, кражу личных данных и установку вредоносного программного обеспечения.

2. Интранет

Интранет – это внутренняя сеть, специально разработанная для управления информацией внутри компании или, например, частной домашней сети. Интранет является единым пространством для хранения, обмена и доступа к информации для всех компьютеров сети. Поэтому, если какой-либо из компьютеров сети заражен, остальные компьютеры подвергаются огромному риску заражения. Во избежание возникновения таких ситуаций необходимо защищать не только периметр сети, но и каждый отдельный компьютер.

3. Электронная почта

Наличие почтовых приложений практически на каждом компьютере, а также то, что вредоносные программы полностью используют содержимое электронных адресных книг для выявления новых жертв, обеспечивает благоприятные условия для распространения вредоносных программ.

Пользователь зараженного компьютера, сам того не подозревая, рассылает зараженные письма адресатам, которые в свою очередь отправляют новые зараженные письма и т. д. Нередки случаи, когда зараженный файл-документ по причине недосмотра попадает в списки рассылки коммерческой информации какой-либо крупной компании. В этом случае страдают не пять, а сотни или даже тысячи абонентов таких рассылок, которые затем разошлют зараженные файлы десяткам тысячам своих абонентов.

Помимо угрозы проникновения вредоносных программ существует проблема внешней нежелательной почты рекламного характера (спама). Не являясь источником прямой угрозы, нежелательная корреспонденция увеличивает нагрузку на почтовые сервера, создает дополнительный трафик, засоряет почтовый ящик пользователя, ведет к потере рабочего времени и тем самым наносит значительный финансовый урон.

4. Съёмные носители информации

Съёмные носители – дискеты, CD-диски, флеш-карты – широко используются для хранения и передачи информации.

При запуске файла, содержащего вредоносный код, со съёмного носителя вы можете повредить данные, хранящиеся на вашем компьютере, а также распространить вирус на другие диски компьютера или компьютеры сети.

Виды угроз вычислительной техники

В настоящее время существует огромное количество угроз, которым может подвергнуться ваш компьютер.

1) Черви (Worms)

Данная категория вредоносных программ для распространения использует в основном уязвимости операционных систем. Название этого класса было дано исходя из способности червей «переползать» с компьютера на компьютер, используя сети, электронную почту и другие информационные каналы. Также благодаря этому многие черви обладают достаточно высокой скоростью распространения.

Черви проникают на компьютер, вычисляют сетевые адреса других компьютеров и рассылают по этим адресам свои копии. Помимо сетевых адресов часто используются данные адресной книги почтовых клиентов. Представители этого класса вредоносных программ иногда создают рабочие файлы на дисках системы, но могут вообще не обращаться к ресурсам компьютера (за исключением оперативной памяти).

2) Вирусы (Viruses)

Программы, которые заражают другие программы – добавляют в них свой код, чтобы получить управление при запуске зараженных файлов. Это простое определение дает возможность выявить основное действие, выполняемое вирусом – заражение. Таким образом, программа, которая поражена вирусом, будет запускаться, однако или в необычном виде (например, с повернутым на 180 градусов экраном), или с добавлением в

функции каких-то дополнительных особенностей (текст, посвященный борьбе с атомной энергетикой у вируса Чернобыль) и тому подобное.

3) *Троянские программы (Trojans)*

Программы, которые выполняют на поражаемых компьютерах несанкционированные пользователем действия, т. е. в зависимости от каких-либо условий уничтожают информацию на дисках, приводят систему к «зависанию», воруют конфиденциальную информацию и т. д.

Данный класс вредоносных программ не является вирусом в традиционном понимании этого термина (т. е. не заражает другие программы или данные); троянские программы не способны самостоятельно проникать на компьютеры и распространяются злоумышленниками под видом «полезного» программного обеспечения. При этом вред, наносимый ими, может во много раз превышать потери от традиционной вирусной атаки.

В последнее время наиболее распространенными типами вредоносных программ, портящими компьютерные данные, стали черви. Далее по распространенности следуют вирусы и троянские программы. Некоторые вредоносные программы совмещают в себе характеристики двух или даже трех из перечисленных выше классов.

4) *Программы-рекламы (Adware)*

Программный код, без ведома пользователя включенный в программное обеспечение с целью демонстрации рекламных объявлений. Как правило, программы-рекламы встроены в программное обеспечение, распространяющееся бесплатно. Реклама располагается в рабочем интерфейсе. Зачастую данные программы также собирают и переправляют своему разработчику персональную информацию о пользователе, изменяют различные параметры браузера (стартовые и поисковые страницы, уровни безопасности и т. д.), а также создают неконтролируемый пользователем трафик. Все это может привести как к нарушению политики безопасности, так и к прямым финансовым потерям.

5) *Программы-шпионы (Spyware)*

Программное обеспечение, позволяющее собирать сведения об отдельно взятом пользователе или организации без их ведома. О наличии программ-шпионов на своем компьютере вы можете и не догадываться. Как правило, целью программ-шпионов является:

- отслеживание действий пользователя на компьютере;
- сбор информации о содержании жесткого диска; в этом случае чаще всего речь идет о сканировании некоторых каталогов и системного реестра с целью составления списка программного обеспечения, установленного на компьютере;
- сбор информации о качестве связи, способе подключения, скорости модема и т. д.

6) Потенциально опасные приложения (Riskware)

Программное обеспечение, которое не имеет какой-либо вредоносной функции, но может быть использовано злоумышленниками в качестве вспомогательных компонентов вредоносной программы, поскольку содержит бреши и ошибки. При некоторых условиях наличие таких программ на компьютере подвергает ваши данные риску. К таким программам относятся, например, некоторые утилиты удаленного администрирования, программы автоматического переключения раскладки клавиатуры, IRC-клиенты, FTP-сервера, всевозможные утилиты для останова процессов или скрытия их работы.

7) Программы-шутки (Jokes)

Программное обеспечение, не причиняющее компьютеру какого-либо прямого вреда, но выводящее сообщения о том, что такой вред уже причинен, либо будет причинен при каких-либо условиях. Такие программы часто предупреждают пользователя о несуществующей опасности, например, выводят сообщения о форматировании диска (хотя никакого форматирования на самом деле не происходит), обнаруживают вирусы в незараженных файлах и т. д.

8) Программы-маскировщики (Rootkit)

Утилиты, используемые для сокрытия вредоносной активности. Они маскируют вредоносные программы, чтобы избежать их обнаружения антивирусными программами. Программы-маскировщики модифицируют операционную систему на компьютере и заменяют основные ее функции, чтобы скрыть свое собственное присутствие и действия, которые предпринимает злоумышленник на зараженном компьютере.

Одним из основных методов борьбы с компьютерными угрозами является, как и в медицине, своевременная профилактика. Компьютерная профилактика состоит из небольшого количества правил, соблюдение которых значительно снижает вероятность заражения вирусом и потери каких-либо данных.

Ниже перечислены **основные правила безопасности**, выполнение которых позволит вам избегать вирусных атак.

Правило № 1: защитите ваш компьютер с помощью антивирусных программ и программ безопасной работы в Интернете. Для этого:

1. Безотлагательно установите антивирус.
2. Регулярно обновляйте сигнатуры угроз, входящие в состав программы. Обновление можно проводить несколько раз в день при возникновении вирусных эпидемий – в таких ситуациях новые сигнатуры угроз находятся серверах обновлений.
3. Задайте рекомендуемые экспертами параметры защиты вашего компьютера. Постоянная защита начинает действовать сразу после включения компьютера и затрудняет вирусам проникновение на компьютер.

Правило № 2: будьте осторожны при записи новых данных на компьютер:

Проверяйте на присутствие вирусов все съемные диски (дискеты, CD-диски, флеш-карты и пр.) перед их использованием.

Осторожно обращайтесь с почтовыми сообщениями. Не запускайте никаких файлов, пришедших по почте, если вы не уверены, что они действительно должны были прийти к вам, даже если они отправлены вашими знакомыми.

Внимательно относитесь к информации, получаемой из Интернета. Если с какого-либо веб-сайта вам предлагается установить новую программу, обратите внимание на наличие у нее сертификата безопасности.

Если вы копируете из Интернета или локальной сети исполняемый файл, обязательно проверьте его с помощью Антивируса.

Правило № 3: внимательно относитесь к информации о вирусах в Интернете.

В большинстве случаев в Интернете сообщают о начале новой эпидемии задолго до того, как она достигнет своего пика. Вероятность заражения в этом случае еще невелика, и, скачав обновленные сигнатуры угроз, вы сможете защитить себя от нового вируса заблаговременно.

Правило № 4: с недоверием относитесь к вирусным мистификациям – программам-шуткам, письмам об угрозах заражения.

Правило № 5: пользуйтесь сервисом Windows Update и регулярно устанавливайте обновления операционной системы Microsoft Windows.

Правило №6: покупайте дистрибутивные копии программного обеспечения у официальных продавцов.

Правило № 7: ограничьте круг людей, допущенных к работе на вашем компьютере.

Правило № 8: уменьшите риск неприятных последствий возможного заражения:

Своевременно делайте резервное копирование данных. В случае потери данных система достаточно быстро может быть восстановлена при наличии резервных копий. Дистрибутивные диски и флеш-карты и другие носители с программным обеспечением и ценной информацией должны храниться в надежном месте.

Обязательно создайте диск аварийного восстановления, с которого при необходимости можно будет загрузиться, используя «чистую» операционную систему.

Правило № 9: регулярно просматривайте список установленных программ на вашем компьютере. Таким образом, вы можете обнаружить программное обеспечение, которое было установлено на компьютер без вашего ведома, пока вы, например, пользовались Интернетом или устанавливали некоторую программу. Некоторые из них могут оказаться потенциально опасными программами [3].

Практическая работа № 4

Анализ примеров нарушений информационной безопасности

Порядок выполнения работы

Выявление значимых составляющих информационной безопасности и характеристик информационной системы в конкретных ситуациях

Цель работы: проводить анализ нарушений информационной безопасности.

Теоретические сведения

Информационная безопасность в самом широком смысле – это совокупность средств защиты информации от случайного или преднамеренного воздействия. Независимо от того, что лежит в основе воздействия: естественные факторы или причины искусственного характера – владелец информации несет убытки.

Принципы информационной безопасности

Целостность информационных данных означает способность информации сохранять изначальный вид и структуру как в процессе хранения, как и после неоднократной передачи. Вносить изменения, удалять или дополнять информацию вправе только владелец или пользователь с легальным доступом к данным. Конфиденциальность – характеристика, которая указывает на необходимость ограничить доступ к информационным ресурсам для определенного круга лиц. В процессе действий и операций информация становится доступной только пользователям, который включены в информационные системы и успешно прошли идентификацию. Доступность информационных ресурсов означает, что информация, которая находится в свободном доступе, должна предоставляться полноправным пользователям ресурсов своевременно и беспрепятственно. Достоверность указывает на принадлежность информации доверенному лицу или владельцу, который одновременно выступает в роли источника информации. Обеспечение и поддержка информационной безопасности включают комплекс разноплановых мер, которые предотвращают, отслеживают и устраняют несанкционированный доступ третьих лиц. Меры ИБ направлены также на защиту от повреждений, искажений, блокировки или копирования информации. Принципиально, чтобы все задачи решались одновременно, только тогда обеспечивается полноценная, надежная защита.

Методические указания

1. а) Проанализируйте данные нарушения:

Сотрудники сидят в Интернете на развлекательных сайтах по 4 часа. Играют в игры на работе. Перекалывают свои обязанности на других. Сотрудники копируют конфиденциальные данные компании. Сотрудники одновременно работают на другой работе. Сотрудники работают на конкурента. Сотрудник хочет создать собственную конкурентную компанию. Сотрудники распространяют государственную тайну.

б) Используется потенциально опасное ПО, содержащее уязвимости, позволяющие скрытую передачу данных, дающие возможность деструктивного влияния на ИТ инфраструктуру компании. Выясняется, что административные пароли передаются по открытым каналам. Сотрудники используют средства удаленного управления. Администраторы сделали бреши в межсетевом периметре компании для удобства. Оставлены закладки в самописном программном обеспечении.

2. Как предотвратить потерю информации?
3. Что следует предпринять для устранения данных нарушений?

Контрольные вопросы

1. Что называют информационной безопасностью?
2. Какие составляющие информационной безопасности должны быть соблюдены?
3. Что будет, если одна из составляющих информационной безопасности будет нарушена?

ТЕМА 5. АЛГОРИТМИЧЕСКИЕ ЯЗЫКИ. ОБЩИЕ СВЕДЕНИЯ О ЯЗЫКЕ ПАСКАЛЬ

Понятие алгоритма такое же основополагающее для информатики, как и понятие информации. Именно поэтому важно в нем разобраться.

Алгоритм – это система формальных правил, однозначно приводящая к решению данной задачи. Если это определение сузить до области применения вычислительной техники, то можно определить алгоритм как последовательность арифметических и логических действий над числовыми значениями переменных, приводящую к вычислению решения задачи при изменениях исходных данных в достаточно широких пределах.

Название «алгоритм» произошло от латинской формы имени величайшего среднеазиатского математика Мухаммеда ибн Муса ал-Хорезми (Algorizmi), жившего в 783—850 гг. В своей книге «Краткая книга восполнения и противопоставления» он систематизировал и изложил два известных ему выдающихся достижения индийских математиков: арифметику в позиционной десятичной системе счисления и решение квадратного уравнения. В XII в. эта книга была переведена на латынь и получила широкое распространение в Европе. Латинский перевод книги начинается словами «Dixit Algorizmi» (так сказал Алгоризми). Имя автора в латинизированной форме стало обозначать в средневековой Европе всю систему десятичной арифметики, а позже – и правила выполнения определенных действий, то есть алгоритмы.

Исполнитель алгоритма – это некоторая абстрактная или реальная (техническая, биологическая или биотехническая) система, способная выполнить действия, предписываемые алгоритмом. Исполнителя характеризуют:

- среда;
- элементарные действия;
- система команд;
- отказы.

Каждый исполнитель может выполнять команды только из некоторого строго заданного списка – системы команд исполнителя. Для каждой команды должны быть заданы условия применимости (в каких состояниях среды может быть выполнена команда) и описаны результаты выполнения команды. После вызова команды исполнитель совершает соответствующее элементарное действие.

Отказы исполнителя возникают, если команда вызывается при недопустимом для нее состоянии среды. Обычно исполнитель ничего не знает о цели алгоритма. Он выполняет все полученные команды, не задавая вопросов «почему» и «зачем». В информатике универсальным исполнителем алгоритмов является компьютер. Средой является любая система программирования.

Как и любое научное понятие, алгоритм обладает некоторыми свойствами, характеризующими его выполнение.

Свойства алгоритма:

1. Детерминированность (определенность).

Каждое правило алгоритма должно быть четким, однозначным и не оставлять места для произвола. Благодаря этому свойству выполнение алгоритма носит механический характер и не требует никаких дополнительных указаний или сведений о решаемой задаче. Поэтому применение алгоритма к одним и тем же исходным данным должно приводить к одному и тому же результату.

2. Результативность (конечность).

За конечное число шагов алгоритм либо должен приводить к решению задачи, либо после конечного числа шагов останавливаться из-за невозможности получить решение с выдачей соответствующего сообщения, либо неограниченно продолжаться в течение времени, отведенного для исполнения алгоритма, с выдачей промежуточных результатов.

3. Массовость.

Алгоритм решения задачи разрабатывается в общем виде, т. е. он должен быть применим для некоторого класса задач, различающихся лишь исходными данными. При этом исходные данные могут выбираться из некоторой области, которая называется областью применимости алгоритма.

4. Дискретность (прерывность, раздельность).

Алгоритм должен представлять процесс решения задачи как последовательное выполнение простых (или ранее определенных) этапов (шагов).

5. Понятность.

Исполнитель алгоритма должен понимать, как его выполнять. Другими словами, при наличии алгоритма и произвольного варианта исходных данных исполнитель должен знать, как надо действовать для выполнения этого алгоритма.

Для составления алгоритма и программы решения задач на компьютере математическая формулировка задачи, включающая символы математического анализа (символы интеграла, производной, дифференциальных операторов и т. д.), должна быть преобразована непосредственно в процедуру решения задачи, представляющую собой последовательность арифметических действий и логических связей между ними, то есть должен быть выбран метод решения задачи. Например, при решении квадратного уравнения $ax^2 + bx + c = 0$ вычисление корней производится по формуле

$$x_{1,2} = (-b \pm \sqrt{b^2 - 4ac}) / (2a)$$

Формулы, подобные данной, могут составить основу алгоритмов решения подобных задач.

Недостаток формул заключается в том, что они не обладают достаточной массовостью. Для алгебраических уравнений степени $n \geq 5$ не существует общих формул, выражающих связь между коэффициентами уравнений и их корнями в

радикалах. Поэтому в вычислительной математике разработаны методы арифметизации задач, называемые численными методами, которые приспособлены для решения широких классов задач и обеспечивают их достаточно высокую точность решения. Выбор того или иного численного метода для решения задачи на компьютере связан, с одной стороны, с требованиями, предъявляемыми постановкой задачи (точность решения, быстрота получения результата, стоимость подготовки программы решения задачи), и с другой стороны, с требованиями, предъявляемыми компьютером и программой к численному методу для его компьютерной реализации.

Алгоритм, на основе которого составляется программа решения задачи, представляет собой последовательность действий, которые может выполнить (или может выполнить наиболее рационально) компьютер. В нем отражены не только арифметические действия, необходимые для реализации выбранного численного метода, но и логические связи, которые численный метод налагает на исходные данные. Эти логические связи должны быть заданы в форме, воспринимаемой вычислительными машинами, проверкой тех или иных соотношений, допускающих (или не допускающих) автоматическое выполнение действий, предписываемых машине программой.

Алгоритм может быть представлен различными способами. На практике наиболее распространены следующие формы представления алгоритма:

- словесная (запись на естественном языке);
- графическая (изображения из графических символов);
- псевдокоды (полуформализованные описания алгоритмов на условном алгоритмическом языке, включающие в себя как элементы языка программирования, так и фразы естественного языка, общепринятые математические обозначения и др.);
- программная (тексты на языках программирования).

При графическом представлении алгоритм изображается в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению одного или нескольких действий.

Все символы схем алгоритмов описываются государственным стандартом ГОСТ 19.701.90 «Схемы алгоритмов, программ, данных и систем», в котором точно обозначены размеры каждого блока, его вид и назначение.

Алгоритмический язык программирования — это язык, специально предназначенный для написания алгоритмов и программ. Он обладает набором команд и синтаксических правил, которые позволяют программисту описывать последовательность операций для выполнения определенной задачи. Язык программирования Паскаль был создан швейцарским профессором кибернетики Никлаусом Виртом (1934–2024) специально для обучения студентов программированию.

Алфавит Pascal включает буквы, цифры, шестнадцатеричные цифры, специальные символы, пробелы и зарезервированные слова.

Буквы – это буквы латинского алфавита от а до z и от А до Z, а также знак подчеркивания `_`. Нет различия между прописными и строчными буквами алфавита, если только они не входят в символьные или строковые выражения.

Цифры – арабские цифры от 0 до 9.

Каждая шестнадцатеричная цифра имеет значение от 0 до 15. Первые 10 значений обозначаются арабскими цифрами 0..9, остальные шесть – латинскими A..F.

Специальные символы – это символы + - * / = , ' . : ; , . [] () { } ^ @ \$ #

К специальным также относятся следующие пары символом: < > <= >= := (* *) (.)

Зарезервированные (служебные) слова: and, asm, array, begin, case, const, construction, destructor, div, do, downto, else, end, file, for, function, goto, if, implementation, in, inline, interface, label, mod, nil, not, object, of, or, packed, procedure, program, record, repeat, set, shl, shr, string, then, to, type, unit, until, uses, var, while, with, xor.

Идентификаторы – имена констант, переменных, меток, типов, объектов, процедур, функций, модулей, программ и т. д. В качестве идентификатора можно использовать любые последовательности символов, удовлетворяющих следующим ограничениям:

1) идентификатор может состоять из букв латинского алфавита, цифр, знака подчеркивания; другие символы в идентификаторе не допустимы;

2) идентификатор не может начинаться с цифры;

3) идентификатор не может совпадать ни с одним из зарезервированных слов;

4) длина идентификатора может быть произвольной, но значащими считаются первые 63 символа.

Константы. В качестве констант могут использоваться целые, вещественные, шестнадцатеричные числа, логические константы, символы, строки символов, конструкторы множеств и признак неопределенного указателя NIL.

Тип константы определяется способом записи ее значения. Например:

Const

c1 = 17; (константа типа integer)

c2 = 3.14; (константа типа real)

c3 = 'A'; (константа типа char)

c4 = '3.14'; (константа типа string)

c5 = false; (константа типа boolean)

Целые числа записываются со знаком или без него по обычным правилам, могут принимать значения от -2147483648 до +2147483847.

Вещественные числа записываются со знаком или без него с использованием десятичной точки и/или экспоненциальной части (e или E за которой следуют «+» или «-» и десятичный порядок – например, 3.14e-2 = 3.14*10⁻²).

Шестнадцатеричное число – состоит из шестнадцатеричных цифр, которым предшествует знак \$. Диапазон – от \$00000000 до \$FFFFFFF.

Логическая константа – это либо слово FALSE (ложь) либо TRUE (истина).

Символьная константа – это любой символ, заключенный в апострофы (пример – 'z').

Строковая константа – любая последовательность символов, заключенная в апострофы.

Конструктор множества – список элементов множества, обрамленных квадратными скобками (пример [1, 2, 3, 4...7, 12]).

Комментарии разрешается вставлять в любое место программы, где по смыслу может стоять пробел. Комментарии обрамляются ограничителями – {комментарий} или (* комментарий *). Можно вкладывать комментарии с ограничителями разных типов друг в друга: { (* *) } или (* { } *).

Синтаксис на примере программы:

```
Program My_first_program;
```

```
Const
```

```
text = 'Ya programmiuyu na Pascale';
```

```
Begin
```

```
writeln(text);
```

```
end.
```

В данной программе-примере 6 строк. Строки выделяют некоторые смысловые фрагменты текста и могут не связываться с конкретными действиями в программе. Пробел в Pascal используется как разделитель отдельных конструкций языка и не может использоваться произвольно. Поэтому в названии программы (My_first_program) используются символы подчеркивания. В языке игнорируется различие по высоте букв (заглавные и строчные), если только это не связано с текстовыми константами.

Первая строка – слово **Program** зарезервировано и содержит объявление имени программы. Зарезервированные слова не могут использоваться в качестве идентификатора (имени) объекта программы – переменной, константы и т. д. Вторая строка – **const** – содержит зарезервированное слово **const**, означающее, что далее будут описываться одна или несколько констант. Константы – такие объекты программы, которые не могут изменять свои значения. Описать константу – значит указать ее имя и значение (третья строка: text – имя константы, 'Ya programmiuyu na Pascale' – значение константы). text – константа типа строки символов (признаком этого являются апострофы). Четвертая строка – зарезервированное слово **begin** – начало главной части программы (раздела операторов). Пятая строка: оператор writeln(text) – вывод значения константы text на экран. Слово writeln не является зарезервированным, но является средством вывода данных, поэтому переопределять его не стоит. Шестая строка: зарезервированное слово **end** с точкой – конец программы. За сочетанием **end.** может идти какой угодно текст, он не будет обрабатываться компилятором» [4].

Практическая работа № 5

Язык программирования Паскаль, основные операторы

Порядок выполнения работы

Составить программу на языке Паскаль согласно варианту задания. В отчете необходимо указать цель работы, словесно описать последовательность действий, приводящую к решению поставленной задачи, представить структурную схему (блок-схему) алгоритма, текст созданной программы, контрольный пример (примеры), показывающий выполнение заданной цели.

1. Вычислить длину окружности и площадь круга одного и того же заданного радиуса R .

2. Вычислить расстояние между двумя точками с данными координатами на плоскости (x_1, y_1) и (x_2, y_2) .

3. Дана длина ребра куба. Найти площадь грани, площадь полной поверхности и объем этого куба.

4. Три сопротивления R_1, R_2, R_3 соединены параллельно. Найти сопротивление всей цепи.

5. Найти сумму членов арифметической прогрессии, если известны ее первый член, разность и число членов прогрессии.

6. Вычислить корни квадратного уравнения $ax^2 + bx + c = 0$ с заданными коэффициентами a, b и c (предполагается, что $a \neq 0$ и что дискриминант уравнения неотрицателен).

7. Найти площадь равнобедренной трапеции с основаниями a и b и углом α при большем основании a .

8. Заданы координаты трех вершин треугольника $(x_1, y_1), (x_2, y_2), (x_3, y_3)$. Найти его периметр и площадь.

9. Найти произведение всех цифр заданного четырехзначного числа.

10. Вычислить площадь и периметр правильного N -угольника, описанного около окружности радиуса R (рассмотреть N целого типа, R вещественного типа).

11. Дано натуральное число T – длительность прошедшего времени в секундах. Вывести данное значение длительности в часах (НН), минутах (ММ) и секундах (SS) в следующей форме: НН ч ММ мин SS с.

12. Дано действительное число R вида nnn.ddd (три цифровых разряда в целой и дробной частях). Поменять местами дробную и целую части числа и вывести полученное значение числа.

13. Составить программу перевода радианной меры угла в градусы, минуты и секунды.

14. С начала суток часовая стрелка повернулась на y градусов ($0 \leq y < 360$, y – вещественное число). Определить число полных часов и полных минут, прошедших с начала суток. Сформулировать и решить обратную задачу.

ТЕМА 6. АРИФМЕТИЧЕСКИЕ ОСНОВЫ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

Системой счисления (СС) называется совокупность приемов наименования и записи чисел. В любой СС для представления чисел выбираются некоторые символы (слова или знаки), называемые базисными числами, а все остальные получаются в результате каких-либо операций из базисных чисел данной системы счисления.

Символы, используемые для записи чисел, могут быть любыми, только – разными, и значение каждого из них должно быть известно.

Наиболее распространено представление в виде арабских цифр $\{0, \dots, 9\}$.

В римской системе счисления базисными являются числа 1, 5, 10, 50, 100, 500, 1000, которые обозначаются знаками I, V, X, L, C, D, M, а все остальные получаются операциями сложения или вычитания базисных:

– если цифра слева меньше, чем цифра справа, то левая цифра вычитается из правой;

– если цифра справа меньше или равна цифре слева, то эти цифры складываются.

Например, число 146 будет записано в римской системе счисления как CXLVI. Подобные системы счисления называются аддитивными. При сложении и вычитании больших чисел, а также при умножении и делении римская СС неудобна. В ней каждый числовой знак в записи любого числа имеет одно и то же значение, которое не зависит от его расположения в записи числа. Поэтому эта система относится к *непозиционным*.

Система называется *позиционной*, если значение каждой цифры (ее вес) изменяется в зависимости от ее положения (позиции) в последовательности цифр, изображающих число.

В десятичной СС каждый разряд имеет вес, равный степени 10:

$$343.32_{(10)} = 3 * 10^2 + 4 * 10^1 + 3 * 10^0 + 3 * 10^{-1} + 2 * 10^{-2} \quad (1)$$

Число K единиц какого-либо разряда, объединяемых в единицу более старшего разряда, называют *основанием позиционной системы счисления*, а сама система счисления называется K -ичной.

Запись произвольного числа в K -ичной позиционной СС основывается на представлении этого числа в виде полинома, где каждый коэффициент a_i может быть одним из базисных чисел и изображается одной цифрой

$$X = a_n * K_n + a_{n-1} * K_{n-1} + \dots + a_1 * K_1 + a_0 * K_0 + a_{-1} * K_{-1} + \dots + a_{-m} * K_{-m} + \dots \quad (2)$$

В современной вычислительной технике, в устройствах автоматики и связи широко используется *двоичная система счисления*. Это система счисления с наименьшим возможным основанием. В ней для изображения числа используются только две цифры: 0 и 1.

Двоичная система счисления имеет ряд преимуществ перед другими системами:

– для ее реализации нужны технические устройства с двумя устойчивыми состояниями (есть ток – нет тока, намагничен – не намагничен и т. п.), а не, например, с десятью, как в десятичной;

– представление информации посредством только двух состояний надежно и помехоустойчиво;

– возможно применение аппарата булевой алгебры для выполнения логических преобразований информации;

– двоичная арифметика намного проще десятичной.

Примеры изображения чисел в двоичной системе счисления

$$0.25_{10} = 0.01_2$$

$$5_{10} = 101_2$$

$$0.5_{10} = 0.1_2$$

$$6_{10} = 110_2$$

$$1_{10} = 1_2$$

$$7_{10} = 111_2$$

$$2_{10} = 10_2$$

$$8_{10} = 1000_2$$

$$3_{10} = 11_2$$

$$9_{10} = 1001_2$$

$$4_{10} = 100_2$$

$$10_{10} = 1010_2$$

Недостаток двоичной системы – быстрый рост числа разрядов, необходимых для записи чисел, т. е. в громоздкости записи чисел, но это не имеет существенного значения для ЭВМ. Однако если возникает необходимость кодировать информацию «вручную», например, при составлении программы на машинном языке, то предпочтительнее оказывается пользоваться восьмеричной или шестнадцатеричной системой счисления.

В восьмеричной системе счисления базовыми являются числа от 0 до 7. Например, десятичное число 83.5 в восьмеричной системе счисления может быть записано следующим образом:

$$83.5_{(10)} = 64 + 16 + 3 + 1/2 = 64 + 16 + 3 + 4/8 = 1*8^2 + 2*8^1 + 3*8^0 + 4*8^{-1} = 123.4_{(8)}$$

В шестнадцатеричной системе счисления базовыми являются числа от 0 до 15. Так как арабских цифр для полной записи числа в шестнадцатеричной системе счисления может не хватать, для их записи используются латинские буквы от А до F. Например, десятичное число 175.5 может быть записано в шестнадцатеричной системе счисления так:

$$175.5_{(10)} = 160 + 15 + 1/2 = 10*16^1 + 15*16^0 + 8*16^{-1} = af.8_{(16)}$$

Иногда числа, заданные в СС с основанием Р, приходится изображать с помощью цифр другой СС с основанием Q, где Q < Р.

Такая ситуация возникает, например, когда в ЭВМ, способной непосредственно воспринимать только двоичные числа, необходимо изобразить десятичные числа, с которыми мы привыкли работать.

В этих случаях используются смешанные системы счисления, в которых каждый коэффициент Р-ичного разложения числа записывается в Q-ичной системе. В такой системе Р называется старшим основанием, Q – младшим, а сама система называется (Q-Р)-ичной.

Например, в смешанной двоично-десятичной системе счисления для изображения каждой десятичной цифры отводится 4 двоичных разряда. Десятичное число 925 в такой системе запишется в виде

$$925_{(10)} = 1001\ 0010\ 0101_{(2-10)}.$$

В общем случае изображение числа x в P -ичной системе счисления в случае $P=Q^l$ является просто сокращенной записью изображения этого же числа x в Q -ичной системе счисления. Это свойство используется для сокращенной записи чисел, заданных в системах счисления с небольшим основанием. Для этого в исходной записи числа разряды объединяются в группы некоторой длины (с добавлением слева или справа необходимого количества нулей), и каждая такая группа записывается одной цифрой другой системы, основание которой равно соответствующей степени исходного основания.

Например, двоичное изображение десятичного числа 46.5, равное 101110.1, можно записать короче с помощью цифр других систем, причем эта сокращенная запись одновременно является и изображением данного числа в соответствующей системе счисления:

10 11 10. 10 = 232.2₍₄₎ (в двоичной записи отделяем слева и справа от точки по ДВА разряда, исходная степень – ВТОРАЯ, 2 в степени 2 = 4 – четверичная система);

101 110. 100 = 56.4₍₈₎ (в двоичной записи отделяем слева и справа от точки по ТРИ разряда, исходная степень – ВТОРАЯ, 2 в степени 3 = 8 – восьмеричная система);

0010 1110. 1000 = 2e.8₍₁₆₎ (в двоичной записи отделяем слева и справа от точки по ЧЕТЫРЕ разряда, исходная степень – ВТОРАЯ, 2 в степени 4 = 16 – шестнадцатеричная система).

Перевод чисел из одной системы счисления в другую происходит по формуле (2):

а) перевести число $x=371_{(8)}$ в десятичную систему счисления –
 $x=371_{(8)}=3*8^2+7*8^1+1*8^0=192+56+1=249_{(10)}$;

б) перевести число $x=af.4_{(16)}$ в десятичную систему счисления –
 $x=af.4_{(16)}=10*16^1+15*16^0+4*16^{-1}=160+15+4/16=175.25_{(10)}$;

в) перевести десятичное число $N=47$ в двоичную систему счисления –
 $47/2=23/2=11/2=5/2=2/2=1/2$

1 1 1 1 0 1

Полученные таким образом остатки от деления записываются справа налево

$N=47_{(10)} = 101111_{(2)} \gg [1]$.

Практическая работа № 6 Перевод чисел в различные системы счисления

Порядок выполнения работы

Выполнить указанные задания по вариантам, обязательно показав в представленном отчете, каким образом был осуществлен перевод чисел.

Вариант 1.

1. Переведите числа из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления.

- а) 948;
- б) 763;
- в) 994,125;
- г) 523,25;
- д) 203,82.

2. Переведите числа в десятичную систему счисления.

- а) 111000111_2 ;
- б) 100011011_2 ;
- в) $1001100101,1001_2$;
- г) $1001001,011_2$;
- д) 335,7₈;
- е) 14C, A₁₆.

3. Выполните сложение чисел.

- а) $1110101010_2 + 10111001_2$;
- б) $10111010_2 + 10010100_2$;
- в) $111101110,1011_2 + 1111011110,1_2$;
- г) 1153,2₈ + 1147,32₈;
- д) 40F,4₁₆ + 160,4₁₆.

4. Выполните вычитание чисел.

- а) $1000000100_2 - 101010001_2$;
- б) $1010111101_2 - 111000010_2$;
- в) $1101000000,01_2 - 1001011010,011_2$;
- г) 2023,5₈ - 527,4₈;
- д) 25E,6₁₆ - 1B1,5₁₆.

5. Выполните умножение чисел.

- а) $1001011_2 * 1010110_2$;
- б) 1650,2₈ * 120,2₈;
- в) 19,4₁₆ * 2F,8₁₆.

Вариант 2

1. Переведите числа из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления.

- а) 563;
- б) 264;
- в) 234,25;

- г) 53,125;
- д) 286,16.

2. Переведите числа в десятичную систему счисления.

- а) 1100010010_2 ;
- б) 10011011_2 ;
- в) $1111000001,01_2$;
- г) $10110111,01_2$;
- д) 416,1₈;
- е) $215,7_{16}$.

3. Выполните сложение чисел.

- а) $10111111_2 + 110010000_2$;
- б) $110010100_2 + 1011100001_2$;
- в) $1000000101,0101_2 + 1010000110,01_2$;
- г) $1512,4_8 + 1015,2_8$;
- д) $274,5_{16} + DD,4_{16}$.

4. Выполните вычитание чисел.

- а) $1000001001_2 - 111110100_2$;
- б) $1111000101_2 - 1100110101_2$;
- в) $1100110101,1_2 - 1011100011,01_2$;
- г) $1501,34_8 - 1374,5_8$;
- д) $12D,3_{16} - 39,6_{16}$.

5. Выполните умножение чисел.

- а) $111101_2 * 1010111_2$;
- б) $1252,14_8 * 76,04_8$;
- в) $66,68_{16} * 1E,3_{16}$.

Вариант 3

1. Переведите числа из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления.

- а) 279;
- б) 281;
- в) 841,375;
- г) 800,3125;
- д) 208,92.

2. Переведите числа в десятичную систему счисления.

- а) 1100111001_2 ;
- б) 10011101_2 ;
- в) $1111011,001_2$;
- г) $110000101,01_2$;
- д) 1601,56₈;
- е) $16E,В4_{16}$.

3. Выполните сложение чисел.

- а) $1000100001_2 + 1011100110_2$;
- б) $1101110011_2 + 111000101_2$;
- в) $1011011,01_2 + 1000101110,1001_2$;

- г) $665,1_8 + 1217,2_8$;
д) $30C,7_{16} + 2A1,8_{16}$.

4. Выполните вычитание чисел.

- а) $11110010_2 - 10101001_2$;
б) $1110100001_2 - 1011001001_2$;
в) $1101001010,1_2 - 1011101001,11011_2$;
г) $166,14_8 - 143,2_8$;
д) $287,A_{16} - 62,8_{16}$.

5. Выполните умножение чисел.

- а) $1001001_2 * 100010_2$;
б) $324,2_8 * 122,12_8$;
в) $F,4_{16} * 38,6_{16}$.

Вариант 4

1. Переведите числа из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления.

- а) 737;
б) 92;
в) 934,25;
г) 413,5625;
д) 100,94.

2. Переведите числа в десятичную систему счисления.

- а) 1110000010_2 ;
б) 1000100_2 ;
в) $110000100,001_2$;
г) $1001011111,00011_2$;
д) $665,42_8$;
е) $246,18_{16}$.

3. Выполните сложение чисел.

- а) $11110100_2 + 110100001_2$;
б) $1101110_2 + 101001000_2$;
в) $1100110011,1_2 + 111000011,101_2$;
г) $1455,04_8 + 203,3_8$;
д) $14E,8_{16} + 184,3_{16}$.

4. Выполните вычитание чисел.

- а) $1000010101_2 - 100101000_2$;
б) $1001011011_2 - 101001110_2$;
в) $11111011,101_2 - 100000010,01_2$;
г) $341,2_8 - 275,2_8$;
д) $249,5_{16} - EE,A_{16}$.

5. Выполните умножение чисел.

- а) $1001000_2 * 1010011_2$;
б) $412,5_8 * 13,1_8$;
в) $3B,A_{16} * 10,4_{16}$.

ТЕМА 7. ЛОГИЧЕСКИЕ ОСНОВЫ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

Познание истины, получение истинной информации – одна из важнейших потребностей человека. Все люди нуждаются в истинном знании, получении новой верной информации о мире, в котором они живут.

Человек с древних времен стремился познать законы правильного мышления, то есть логические законы. Наука логика помогает познанию этих законов. Законы развития есть у природы, общества, любой сложной системы и, конечно же, у самого мышления. Мы можем не осознавать их, но нужно всегда следовать им, чтобы жить в обществе, общаться с людьми, понимать их и быть понятыми.

В Древней Греции, в Древней Индии, в Древнем Риме законы и формы правильного мышления изучались в рамках ораторского искусства. Применение логических приемов рассуждения позволяло ораторам более убедительно доносить до аудитории свою точку зрения, склонять ее на свою сторону. Однако уже тогда возникли попытки использования приемов рассуждения, нарушающих логические законы, чтобы добиться победы в споре.

Логика – одна из древнейших наук. Ее основателем считается величайший древнегреческий философ Аристотель, который первым систематизировал формы и правила мышления, обстоятельно исследовал категории «понятие» и «суждение», подробно разработал теорию умозаключений и доказательств, описал ряд логических операций, сформулировал основные законы мышления.

Античную логику, основанную Аристотелем, принято называть формальной. Это название происходит от основного принципа такой логики как науки, который гласит, что правильность рассуждения определяется только его логической формой или структурой и не зависит от конкретного содержания входящих в него высказываний. В современной логике этот принцип не всегда выдерживается.

Логической формой высказывания (суждения) является строение этого высказывания, способ связи его составных частей.

Идею о возможности математизации логики высказал еще в XVII веке Г. В. Лейбниц. Он пытался создать универсальный язык, с помощью которого каждому понятию и высказыванию можно было бы дать числовую характеристику и установить такие правила оперирования с этими числами, которые позволили бы сразу определить, истинно данное высказывание или ложно. То есть споры между людьми можно было бы разрешать посредством вычислений. Идея Лейбница оказалось ложной, так как невозможно (не найдены способы) свести человеческое мышление к некоторому математическому исчислению.

Однако подлинный прогресс этой науки был достигнут в середине XIX века прежде всего благодаря трудам Джорджа Буля «Математический анализ логики». Он перенес на логику законы и правила алгебраических действий, ввел логические операции, предложил способ записи высказываний в символической форме.

Современная математизированная формальная логика представляет собой обширную научную область, которая находит широкое применение как внутри математики (исследование оснований математики), так и вне ее (синтез и анализ автоматических устройств, теоретическая кибернетика, в частности, искусственный интеллект).

Алгебра логики (алгебра высказываний) – раздел математической логики, изучающий строение (форму, структуру) сложных логических высказываний и способы установления их истинности с помощью алгебраических методов [5].

Под высказыванием (суждением) будем понимать повествовательное предложение, относительно которого можно сказать, истинно оно или ложно.

Обозначать высказывания будем заглавными буквами. Если высказывание A истинное, то будем писать « $A = 1$ » и говорить: « A – истинно». Если высказывание X ложно, то будем писать « $X = 0$ » и говорить: « X – ложно».

ПРИМЕРЫ

A = «Солнце светит для всех» = 1 – истинное высказывание.

B = «Все ученики любят информатику» = 0 – ложное высказывание.

C = «Некоторые из учеников любят информатику» = 1.

D = « A ты любишь информатику?» – не высказывание, т. к. не является повествовательным предложением.

E = «Посмотри в окно» – не высказывание, т. к. является побудительным предложением.

$Ж$ = « $X * X < 0$ » = 0 – ложное высказывание, т. к. какое бы X мы не взяли, произведение $X * X$ будет неотрицательным.

$З$ = « $2 * X - 5 > 0$ » – не высказывание, т. к. для некоторых значений X это выражение будет верным, а для других – нет.

$И$ = «Крокодилы летают очень низко» – высказывание.

Последний пример показывает, что высказывания не обязательно должны быть истинными с точки зрения здравого смысла. Вопрос о том, летают или не летают крокодилы, могут волновать зоологов, но никак не логиков, так как им этот факт безразличен. Логическая наука интересуется весьма своеобразно понимаемой истинностью или ложностью высказываний: под логическим высказыванием понимается определенное утверждение, которое может быть доказано или опровергнуто.

В алгебре логики над высказываниями можно производить различные операции (как и в алгебре действительных чисел, определены операции сложения, деления, возведения в степень над числами).

1. Логическое отрицание (инверсия)

Образуется из простого высказывания с помощью добавления частицы НЕ к сказуемому или использованием оборота речи «НЕВЕРНО, ЧТО ...».

A	Значение A	инверсия A	Значение не A
$У$ меня есть приставка «DENDY»	0	$У$ меня нет приставки «DENDY»	1
$Я$ не знаю китайский язык	1	Неверно, что $я$ не знаю китайский язык ($я$ знаю китайский язык)	0

Инверсия обозначается: не A ; \bar{A} ; not A

Нас интересует значение истинности высказывания формы не A (а не его содержание). Определяется оно по специальной таблице истинности, которая для операции инверсии выглядит так:

A	\bar{A}	Читается
0	1	если A ложно, то не A истинно
1	0	если A истинно, то не A ложно

Мнемоническое правило: слово *инверсия* (от лат. *inversio* – переворачивание) означает, что белое меняется на черное, добро на зло, красивое на безобразное, истина на ложь, ложь на истину, ноль на один, один на ноль.

Примечание 1. Логики предпочитают иметь дело с выражениями «неверно, что», поскольку тем самым подчеркивается отрицание всего высказывания.

Примечание 2. Дважды или четырежды отрицавшееся высказывание имеет то же самое значение истинности, что и соответствующее не отрицавшееся высказывание, трижды отрицавшееся – что и отрицавшееся один раз.

2. Логическое умножение (конъюнкция)

Образуется соединением двух высказываний в одно с помощью союза «И».

ПРИМЕРЫ: Допустим, из моего окна видна автостоянка, на которой обычно стоят две машины: «Мерседес» и «Жигули», но может находиться и какая-то одна из них, или не быть ни одной. Обозначим высказывания:

A = На автостоянке стоит «Мерседес»

B = На автостоянке стоят «Жигули»

A конъюнкция B = На автостоянке стоят «Мерседес» и «Жигули»

Операция конъюнкции обозначается: \wedge ; &; *; and; и.

A	B	A \wedge B	Пояснение	Стоят «Мерседес» и «Жигули»
0	0	0	«Мерседес» не стоит, «Жигули» не стоят	ЛОЖЬ
0	1	0	«Мерседес» не стоит, «Жигули» стоят	ЛОЖЬ
1	0	0	«Мерседес» стоит, «Жигули» не стоят	ЛОЖЬ
1	1	1	«Мерседес» стоит, «Жигули» стоят	ИСТИНА

Из таблицы истинности следует, что операция конъюнкции истинна тогда и только тогда, когда оба высказывания истинны, и ложна, когда хотя бы одно высказывание ложно. Иногда это свойство принимают за определение операции логического умножения.

3. Логическое сложение (дизъюнкция)

Образуется соединением двух высказываний в одно с помощью союза ИЛИ.

Примеры

Завтра дождь будет или не будет (третьего не дано).

Петя сидит на западной или восточной трибуне стадиона.

Студент едет в электричке или читает книгу.

Обозначается:

A или B; A OR B; A | B; A \vee B

ПРИМЕРЫ: допустим, из моего окна видна автостоянка, на которой обычно стоят две машины: «Мерседес» и «Жигули», но может находиться и какая-то одна из них, или не быть ни одной. Обозначим высказывания:

A = На автостоянке стоит «Мерседес»

B = На автостоянке стоят «Жигули»

A дизъюнкция B = На автостоянке стоят «Мерседес» или «Жигули»

A	B	A \vee B	Пояснение	Стоят «Мерседес» или «Жигули»
0	0	0	«Мерседес» не стоит, «Жигули» не стоят	ЛОЖЬ
0	1	1	«Мерседес» не стоит, «Жигули» стоят	ИСТИНА
1	0	1	«Мерседес» стоит, «Жигули» не стоят	ИСТИНА
1	1	1	«Мерседес» стоит, «Жигули» стоят	ИСТИНА

Из таблицы истинности следует, что операция дизъюнкции ложна тогда и только тогда, когда оба высказывания ложны, и истинна, когда хотя бы одно высказывание истинно. Иногда это свойство принимают за определение операции логического умножения.

4. Логическое следование (импликация)

Образуется соединением двух высказываний в одно с помощью оборота речи «ЕСЛИ ..., ТО...»

ПРИМЕРЫ

Если клятва дана, то она должна выполняться.

Если число делится на 9, то оно делится на 3.

Исторически операция импликации была введена для полноты системы логических функций двух переменных, поэтому в логике допустимо (принято, договорились) рассматривать и бессмысленные с житейской точки зрения высказывания. Приведем примеры, которые не только правомерно рассматривать в логике, но при этом значение их истинно.

Если коровы летают, то $2 + 2 = 5$

Если я – Наполеон, то у кошки четыре ноги.

Импликация обозначается: $A \rightarrow B$;

Говорят: «Если A , то B », « A имплицирует B », « A влечет B », « B следует из A ».

A	B	$A \rightarrow B$	Пояснение	«Если идет дождь, то асфальт мокрый»
0	0	1	дождя нет, асфальт сухой	ИСТИНА
0	1	1	дождя нет, асфальт мокрый	ИСТИНА
1	0	0	дождь идет, асфальт сухой	ЛОЖЬ
1	1	1	дождь идет, асфальт мокрый	ИСТИНА

Из таблицы истинности видно, что импликация двух высказываний ложна тогда и только тогда, когда из истинного высказывания следует ложное (истинная предпосылка ведет к ложному выводу). Иногда это свойство принимают за определение операции импликации.

5. Логическое равенство (эквивалентность)

Образуется соединением двух высказываний в одно при помощи оборота речи «... ТОГДА И ТОЛЬКО ТОГДА, КОГДА ...».

ПРИМЕРЫ

«Угол называется прямым тогда и только тогда, когда он равен 90 градусов»

«Две прямые параллельны тогда и только тогда, когда они не пересекаются»

«Любая материальная точка сохраняет состояние покоя или равномерного прямолинейного движения тогда и только тогда, когда внешнее воздействие не изменит этого состояния» (Первый закон Ньютона).

«Голова думает тогда и только тогда, когда язык отдыхает» (Шутка).

Все законы математики, физики, все определения – суть эквивалентность высказываний.

Эквивалентность обозначается: $A = B$; $A \sim B$

ПРИМЕР. Пусть даны два высказывания:

$A =$ «Число делится на 3 без остатка (кратно трем)».

$B =$ «Сумма цифр числа делится нацело на 3».

A эквивалентно $B =$ «Число делится на 3 без остатка тогда и только тогда, когда сумма цифр данного числа делится нацело на 3».

A	B	$A \sim B$	Пояснение	«Число кратно трем тогда и только тогда, когда сумма цифр кратна трем»
0	0	1	число не кратно трем, сумма цифр не кратна трем	ИСТИНА
0	1	0	число не кратно трем, сумма цифр кратна трем	ЛОЖЬ
1	0	0	число кратно трем, сумма цифр не кратна трем	ЛОЖЬ
1	1	1	число кратно трем, сумма цифр кратна трем	ИСТИНА

Из таблицы истинности следует, что эквивалентность двух высказываний истинна тогда и только тогда, когда оба эти высказывания истинны, или оба ложны. Иногда это свойство принимается за определение операции эквивалентности.

ОСНОВНЫЕ ЗАКОНЫ (РАВНОСИЛЬНОСТИ) АЛГЕБРЫ ЛОГИКИ

При решении логических задач часто приходится упрощать формулы. Упрощение формул в булевой алгебре производится на основе эквивалентных преобразований, опирающихся на основные законы.

Законы логики высказываний – это такие выражения, которым всегда соответствует истинное высказывание, какие бы подстановки значений мы ни делали вместо переменных. В алгебре высказываний логические законы выражаются в виде формул.

1. Закон тождества:

$$A = A$$

– всякая мысль тождественна самой себе, то есть « A есть A », где A – любое высказывание.

2. Закон исключенного третьего:

$$A \vee \bar{A} = 1$$

– в один и тот же момент времени высказывание может быть либо истинным, либо ложным, третьего не дано. Истинно либо А, либо не А.

Например: «Число 123 либо четное, либо нечетное, третьего не дано».

Закон исключенного третьего не является законом, признаваемым всеми логиками в качестве универсального закона логики. Этот закон применяется там, где познание имеет дело с жесткой ситуацией: либо–либо, истина–ложь. Там же, где встречается неопределенность (например, в рассуждениях о будущем), закон исключенного третьего часто не может быть применен.

Рассмотрим следующее высказывание: «Это предложение ложно». Оно не может быть истинным, потому что оно утверждает, что оно ложно. Но оно не может быть и ложным, потому что тогда оно было бы истинным. Это высказывание не истинно и не ложно, а потому нарушается закон исключенного третьего.

Парадокс (греч. paradoxos – неожиданный, странный) возникает из-за того, что предложение ссылается само на себя. Известным парадоксом, который сформулировал в 1901 году британский философ и математик Бертран Рассел, является задача о парикмахере:

«В одном городе парикмахер стрижет волосы всем жителям, кроме тех, кто стрижет себя сам. Кто стрижет волосы парикмахеру?»

В нашей формальной системе нет возможности ввести такое ссылающееся само на себя истолкование, поэтому мы не можем выразить все возможные мысли и доводы.

3. Закон непротиворечия:

$$(A \wedge \bar{A}) = 0$$

– не могут быть одновременно истинными суждение и его отрицание. То есть, если высказывание А – истинно, то его отрицание \bar{A} должно быть ложным (и наоборот). Тогда их произведение будет всегда ложным.

За. $A \wedge \bar{A} = 0$.

Именно эта формула часто используется при упрощении сложных логических выражений.

Иногда этот закон формулируется так: два противоречащих друг другу высказывания не могут быть одновременно истинными.

Например: E = «На Марсе есть жизнь и на Марсе жизни нет»

4. Закон двойного отрицания:

$$\bar{\bar{A}} = A$$

– если отрицать дважды некоторое высказывание, то в результате получается исходное высказывание.

Например: $A = \text{«Неверно, что Матроскин не кот»}$
эквивалентно высказыванию $A = \text{«Матроскин – кот»}$.

СВОЙСТВА КОНСТАНТ

5. $\bar{0} = 1$

6. $\bar{1} = 0$

7. $A \vee 0 = A$

8. $A \wedge 0 = 0$

9. $A \vee 1 = 1$

10. $A \wedge 1 = A$

ЗАКОНЫ ИДЕМПОТЕНТНОСТИ

11. $A \vee A = A$

Отсутствие коэффициентов

12. $A \wedge A = A$

Отсутствие степеней

Сколько бы раз мы ни повторяли «на улице тепло» и «на улице тепло» ни на один градус теплее от этого не станет, аналогично, от повторения «телевизор включен» или «телевизор включен» значение высказывания не меняется.

ЗАКОНЫ КОММУТАТИВНОСТИ

13. $A \vee B = B \vee A$

14. $A \wedge B = B \wedge A$

ЗАКОНЫ АССОЦИАТИВНОСТИ

15. $A \vee (B \vee C) = (A \vee B) \vee C$

16. $A \wedge (B \wedge C) = (A \wedge B) \wedge C$

ЗАКОНЫ ДИСТРИБУТИВНОСТИ

17. $A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$

дизъюнкции относительно конъюнкции

18. $A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$

конъюнкции относительно дизъюнкции

Закон 18 аналогичен дистрибутивному закону в алгебре, а закон 17 аналога не имеет, он справедлив только в логике.

Практическая работа № 7

Логические основы информатики

Порядок выполнения работы

Выполнить задания согласно вариантам:

1. Установите, какие из следующих предложений являются логическими высказываниями, а какие – нет (объясните, почему):

- а) «Солнце есть спутник Земли»;
- б) « $2+3>4$ »;
- в) «Сегодня отличная погода»;
- г) «В романе Л.Н. Толстого «Война и мир» 3 432 536 слов»;
- д) «Санкт-Петербург расположен на Неве»;
- е) «Музыка Баха слишком сложна»;
- ж) «Первая космическая скорость равна 7.8 км/сек»;
- з) «Железо — металл»;
- и) «Если один угол в треугольнике прямой, то треугольник будет тупоугольным»;
- к) «Если сумма квадратов двух сторон треугольника равна квадрату третьей, то он прямоугольный».

2. Приведите примеры истинных и ложных высказываний:

- а) из арифметики; б) из физики; в) из биологии; г) из информатики;
- д) из геометрии; е) из жизни.

3. Сформулируйте отрицания следующих высказываний или высказывательных форм:

- а) «Эльбрус — высочайшая горная вершина Европы»;
- б) « $2 > = 5$ »;
- в) « $10 < 7$ »;
- г) «Все натуральные числа целые»;
- д) «Через любые три точки на плоскости можно провести окружность»;
- е) «Теннисист Кафельников не проиграл финальную игру»;
- ж) «Мишень поражена первым выстрелом»;
- з) «Это утро ясное и теплое»;
- и) «Число n делится на 2 или на 3»;
- к) «Этот треугольник равнобедренный и прямоугольный»;
- л) «На контрольной работе каждый ученик писал своей ручкой».

4. Три девочки – Роза, Маргарита и Анюта представили на конкурс цветоводов корзины выращенных ими роз, маргариток и анютиных глазок. Девочка, вырастившая маргаритки, обратила внимание Розы на то, что ни у одной из девочек имя не совпадает с названием любимых цветов. Какие цветы вырастила каждая из девочек?

5. В некотором царстве-государстве повадился Змей Горыныч разбойничать. Послал царь четырех богатырей погубить Змея, а награду за то обещал великую. Вернулись богатыри с победой и спрашивает их царь: «Так кто же из вас главный победитель, кому достанется царева дочь и полцарства?» Засмутились добры молодцы и ответы дали туманные: Сказал Илья Муромец: «Это все Алеша Попович, царь-батюшка». Алеша Попович возразил: «То был Микула Селянинович». Микула Селянинович: «Не прав Алеша, не я это». Добрыня Никитич: «И не я, батюшка». Подвернулась тут баба Яга и говорит царю: «А прав то лишь один из богатырей, видела я всю битву своими глазами». Кто же из богатырей победил Змея Горыныча?

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Острейковский, В. А. Информатика: учеб. для вузов / В. А. Острейковский. – М.: Высшая школа, 2000. – Текст: непосредственный.
2. Назаров, С. В. Введение в программные системы и их разработку / С. В. Назаров, С. Н. Белоусова и др. – Москва: Национальный Открытый Университет ИНТУИТ, 2016. – 650 с. – ISBN intuit120. – URL: <https://ibooks.ru/bookshelf/362830/reading> (дата обращения: 23.04.2024). – Текст: электронный.
3. Касперский, Е. В. Компьютерное зловредство / Е. В. Касперский. – СПб.: Питер, 2008. – 208 с. – Текст: непосредственный. – ISBN 978-5-91180-631-6.
4. Алгоритмические основы информатики: учебно-методическое пособие для студентов I курса / сост. П. Е. Антонюк. – СПб: СПбГТУРП, 2008 – 72 с. – Текст: непосредственный.
5. Подобед, Д. Г. Основы информатики (базовые материалы для курса лекций): учебное пособие / Д. Г. Подобед, М. В. Подобед, О. В. Подобед. – СПб.: СПбГТУРП, 2010. – 75 с. Режим доступа: <http://nizrp.narod.ru/lekziidlyabibl.htm>. – Текст: электронный.

Учебное издание

Антонюк Петр Евгеньевич

Информатика

Учебно-методическое пособие

Редактор и корректор Е. О. Тарновская
Техн. редактор Е. О. Тарновская

Учебное электронное издание сетевого распространения

Системные требования:
электронное устройство с программным обеспечением
для воспроизведения файлов формата PDF

Режим доступа: http://publish.sutd.ru/tp_get_file.php?id=202016, по паролю.
- Загл. с экрана.

Дата подписания к использованию 27.06.2024 г. Рег. № 5088/24

Высшая школа технологии и энергетики СПбГУПТД
198095, СПб., ул. Ивана Черных, 4.