

Лекция для 11СИ, 11СТ. Ориентирована на тест для специальности “Сервис”. Переработка лекций для Ветовой.

Внимание! В тексте возможно больше количество опечаток, не нарушающих изложения (в системе на настоящий момент невозможна проверка орфографии)

## **1. Основные понятия и методы теории информации и кодирования. Сигналы, данные, информация. Общие характеристики процессов сбора, передачи, обработки и накопления информации.**

Информатика – комплекс знаний о получении, преобразовании, хранении и передаче информации. Примерная структура информатики (Могилев, Пак, Хеннер):

- 1) теоретическая информатика;
- 2) вычислительная техника;
- 3) программирование;
- 4) информационные системы;
- 5) искусственный интеллект.

Различают следующие аспекты информатики:

- 1) социальные;
- 2) правовые;
- 3) этические.

Формальное определение информации дать трудно. Информация передается (и вообще существует) в виде сообщений, которые несут некоторое ее количество.

Для передачи информации должны присутствовать 3 элемента: источник, приемник и канал связи (канал передачи). Передача информации происходит вместе с передачей некоторой энергии в определенной материальной среде. Пример: преподаватель читает лекцию студентам. Источник – преподаватель, приемник – студент, канал передачи – воздушная среда, информация передается вместе с колебаниями воздуха.

Количество информации измеряется в битах. 1 бит – это такое количество информации, которое уменьшает степень неизвестности в 2 раза. Например, при бросании монеты может выпасть Г или Р, когда мы бросим монету и узнаем, что выпало, мы получим 1 бит информации.

Для нахождения количества информации используют два подхода: вероятностный и объемный.

Вероятностный подход применяется в случае, когда все возможные результаты опыта нам известны, но неизвестно, какой из них будет реализован). Если результаты опыта равновероятны, то верна формула Хартли

$$H = \log_2 N$$

здесь  $H$  – количество информации,  $N$  – число возможных результатов. Если результаты неравновероятны, то верна формула Шеннона

$$H = \sum p_i \log_2 (1/p_i)$$

здесь  $p_i$  – вероятность  $i$ -того результата.

Объемный подход состоит в следующем: сообщения передаются в виде набора слов определенного алфавита. Символы алфавита – буквы. Каждая буква несет некоторое элементарное количество информации, количество информации во всем сообщении складывается из этих элементарных порций.

Важнейший пример: двоичный код – его алфавит насчитывает всего две “буквы” – это цифры 0 и 1. Ожидая поступления очередной “буквы” сообщения, мы ждем появления одного из двух символов, когда мы узнаем, какой из них поступил, степень неизвестности уменьшится в 2 раза, значит, один двоичный символ несет ровно 1 бит информации, а количество информации в сообщении равно количеству символов в нем.

Нужно знать следующую терминологию: 8 бит (8 двоичных символов) образуют вместе 1 байт,  $2^{10}=1024$  байт составляют килобайт,  $2^{10}$  килобайт составляют 1 мегабайт,  $2^{10}$  мегабайт составляют 1 гигабайт, дальше- терабайт.

Сообщения могут быть записаны не только в двоичном коде. Для представления букв естественных языков в компьютере служат кодировки. Простейшая из них – 8-битная кодировка ASCII. На один символ отводится 8 бит (1 байт), всего существует  $2^8=256$  двоичных 8-значных (8-разрядных) чисел, столько же и символов можно закодировать в этой кодировке. Она содержит буквы английского алфавита, цифры десятичной системы счисления, знаки препинания, арифметических действий и некоторые специальные символы, вроде \$,@,&,# и т.д. Русских букв в таблице этой кодировки нет. Для того, чтобы представлять русские (а так же чешские, болгарские, арабские и какие угодно другие буквы) существуют локальные кодировки. Основные локальные кодировки, содежращие русские буквы: CP1251 (Windows), KOI8-R (Linux, устаревшая), UTF-8 (Linux, современная). Последняя кодировка относится к семейству Unıcod, на каждый символ отводится 16 бит (2 байта). Сообщения в юникоде занимают больше места в памяти компьютера, чем сообщения в 8-битных кодировках. Unıcod завоевывает популярность – имена файлов в OS Windows хранятся уже в Unıcod, а не в Windows-кодировке (CP1251). Видимо, скоро юникод вытеснит все остальные, и это будет несомненно хорошо – пропадет проблема кодировок.

Важность двоичной кодировки обусловлена возможностью непосредственной технической реализации: низкий уровень напряжения и высокий уровень напряжения соответствуют символами 0 и 1. Бурный расцвет вычислительных машин начался именно с осознания необходимости отказа от десятичной системы счисления и перехода на двоичную систему.

В позиционных системах счисления числа записываются ограниченным набором цифр, количество цифр в этом наборе называется основанием системы счисления, значение цифры определяется ее позицией в записи числа. Позиция цифры в записи числа называется разрядом. В десятичной системе счисления разряды, слева направо (по старшинству), следующие: разряд единиц, разряд десятков, разряд сотен, тысяч, десятков тысяч и т.д.

О правилах перехода между двоичной и десятичной системами – см. Приложение 1.

Кроме двоичной, большую роль играет 16-ричная система, в которой используются 16 цифр: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F. Правила перехода – аналогичны правилам для двоичной системы.

Арифметические операции во всех позиционных системах счисления выполняются **ОДИНАКОВО**.

## **2. Технические средства реализации информационных процессов**

### **2.1 Историческая справка.**

В развитии вычислительных машин различают следующие этапы:

- 1) докомпьютерная эпоха. Механические калькуляторы (арифмометры) и попытки создания механических вычислительных машин, управляемых программой (неосуществленная машина Карла Беббиджа, электромеханическая машина Конрада Цюзе).
- 2) первое поколение компьютеров (примерно 1950-1960). Переключательные элементы – электровакуумные лампы. Наиболее известный представитель – ENIAC, США. Баснословно дорогие, малонадежные, очень неудобные в управлении - “1 задача – 1 программа”, программирование – в машинных кодах, доступно только профессионалам высочайшей квалификации. Доступны только крупнейшим университетам в сильнейших странах мира – США, Европа, СССР, Япония. Производство в единичных экземплярах, обслуживание исключительно военных и государственных заказов.
- 3) второе поколение (примерно 1960-1970). Элементы – полупроводниковые транзисторы. Появление первых операционных систем. 1971 – OS Unix. Массовое производство компьютеров, в том числе – коммерческое.
- 4) третье поколение – с 1970 по наше время. Элементы – микроскопические транзисторы, объединенные в интегральные микросхемы (ИМС), в том числе – сверхбольшие интегральные схемы (СБИС, более 1 млн. элементов). Коммерческое производство огромными (миллионными) партиями делает компьютеры все более дешевыми и доступными персональным пользователям. Появление персональных ЭВМ. Появление OS Windows и OS Linux. Массовое появление прикладных программ. Графический интерфейс (окна, иконки, щелчки мышью).
- 5) некоторые исследователи (в частности, авторы теста) выделяют четвертое поколение, начиная с 1985 года. Какой-то особо четкой грани между 3 и 4 поколениями, однако, обнаружить не удастся. 5-е поколение – неудачная попытка в конце XX в. правительства Японии создать компьютеры, особым образом приспособленные к решению логических задач.

### **2.2 Основные узлы компьютера**

В соответствии с определением дисциплины “Информатика” в устройстве компьютера различают следующие основные узлы: устройства ввода, вывода, передачи и преобразования информации. Кроме того, в устройстве компьютера важны многие вспомогательные устройства, такие, как материнская плата, блок питания, слоты расширения и т.д.

- 1) устройства ввода: клавиатура, координатные устройства: мышь, джойстик, тачпад, трекбол, тачскрин (сенсорный экран, он же является и устройством вывода), привод оптических дисков, дисковод (привод гибких магнитных дисков), сканер, графический планшет для художественных работ, видеокамера, датчики снятия сигналов с измерительных приборов;
- 2) устройства вывода: монитор (со всевозможными разновидностями вроде плазменной панели), видеопроектор, принтер, плоттер (принтер гигантских размеров для печати рекламных плакатов), динамические колонки, устройства управления механизмами – станками, печами, стиральными машинами, роботами и т.д, пишущие приводы оптических дисков;
- 3) устройства хранения:
  - а) энергонезависимые: винчестер (жесткий диск), встроенный или внешний, гибкий магнитный диск, магнитная лента (привод называется стримером, применяется редко, но продолжает развиваться), оптические диски CD, DVD, BlueRay и т.д., флэш-память;
  - б) энергозависимая: оперативная память, процессорный кэш;

- 4) устройства передачи: шины передачи данных между узлами компьютера, для связи между компьютерами: кабель (коаксиальный и витая пара), оптоволоконный кабель, беспроводная связь: WiFi, GPRS, BlueTooth, спутниковая связь – с соответствующими устройствами преобразования сигнала;
- 5) устройство преобразования информации – процессор.

### **3. Программные средства реализации информационных процессов**

Программное обеспечение (ПО) - это совокупность программ.

#### **3.1 Классификации ПО.**

1. По назначению: системное, прикладное, инструментальное.
2. По условиям распространения: коммерческое, свободное, открытое.

Системное ПО: ОС (операционная система), драйвера устройств, утилиты (вспомогательные программы).

Прикладное ПО:

- 1) офис: текстовый процессор, табличный редактор, редактор презентаций, СУБД, переводчик;
- 2) мультимедиа: игры, аудиоплееры, видеоплееры, редакторы изображений;
- 3) предприятие: бухгалтерия, электронный документооборот, аудит;
- 4) интернет: браузер, эл. почта, интернет-пейджер (icq), торрент, IP-телефония (skype);
- 5) научное: компьютерная алгебра, система моделирования, САПР.

Инструментальное ПО: среда разработки ПО, СУБД (Википедия).

Комм. ПО - платное: OS Windows, MS Office, Adobe Photoshop.

Свободное ПО (СПО) - бесплатное: OS Linux, OpenOffice, GIMP.

Открытое ПО: поставляется вместе с кодом, любой пользователь может видоизменять код (OS Linux).

#### **3.2 Операционная система.**

ОС - совокупность программ, управляющих оборудованием и создающих среду (платформу) для работы прикладных программ.

Функции:

- 1) загрузка приложений в ОП и их выполнение;
- 2) управление устройствами ввода-вывода и хранения информации;
- 3) распределение ОП между программами;
- 4) пользовательский интерфейс;
- 5) сетевые операции;
- 6) взаимодействие между процессами;
- 7) разграничение прав доступа пользователей;
- 8) защита ПО и персональных данных от вредоносных действий.

#### **3.3 Основные ОС**

Unix — платная, сервера, высокопроизводительные станции, суперкомпьютеры, специализированные компьютеры, системы управления. Графические средства развиты слабо. Стандарт POSIX.

Windows — платная, персональные компьютеры, сервера (однако многие администраторы жалуются на неустойчивость), развлечения. Хорошо развитая мультимедийная составляющая. Ориентирована на массового пользователя.

Mac OS (Макинтош) — платная, персональные компьютеры, акцент на мультимедиа,

удобство пользователя. Стандарт POSIX.

Linux — свободная, открытая, однако имеются дистрибутивы, включающие проприетарное и даже платное прикладное ПО. Сервера, суперкомпьютеры, персональные компьютеры. Средства мультимедиа отстают в развитии от Windows и Mac OS, многие компьютерные игры недоступны. Стандарт POSIX.

#### **4. Модели решения функциональных и вычислительных задач**

Модель — это искусственно созданный объект, дающий упрощенное представление о реальном объекте, процессе или явлении, отражающий существенные стороны изучаемого объекта с точки зрения цели моделирования.

Задачи, которые мы решаем, по своему назначению можно разделить на две категории:

- вычислительные задачи, целью которых является определение некоторой величины;
- функциональные задачи, предназначенные для создания некоего аппарата, выполняющего некоторые действия, функции.

Классификация видов моделирования.

По цели использования: научный эксперимент, комплексные испытания и производственный эксперимент, оптимизационные модели.

По наличию воздействий на систему: детерминированные, стохастические.

По отношению ко времени: статистические, динамические (дискретные и непрерывные).

По возможности реализации: мысленные, реальные.

По области применения: универсальные, специализированные.

Этапы компьютерного математического моделирования:

- определение целей моделирования;
- определение входных и выходных параметров модели;
- построение математической модели;
- выбор метода исследования математической модели;
- разработка алгоритма, составление и отладка программы для ЭВМ.;
- тестирование программы;
- вычислительный эксперимент.

Более подробная информация в приложении: [Моделирование](#)

#### **5. Алгоритмизация и программирование**

##### **5.1 Алгоритм.**

Определение: А. - это строго детерминированная последовательность действий, описывающая преобразование объекта из начального состояния в конечное, записанная с помощью понятных исполнителю команд (Угринович).

Требования к алгоритмам:

- 1) дискретность (пошаговость)
- 2) детерминированность (определенность)
- 3) понятность для исполнителя
- 4) завершаемость
- 5) результативность
- 6) массовость (применимость к разным наборам данных).

Основные структурные элементы алгоритмов:

- 1) линейный алгоритм: ...-команда1-команда2-...
- 2) ветвление (условный переход):  
...-если(условие)-то(набор команд )-иначе(набор команд )-...
- 3) цикл:
  - а) со счетчиком (выполняется до тех пор, пока счетчик не дойдет до некоторого значения);
  - б) с условием (выполняется до тех пор, пока верно условие):
    - б1) с предусловием (условие проверяется в начале цикла);
    - б2) с постусловием (условие проверяется в конце цикла).

## 5.2 Средства записи алгоритмов

1. Псевдокод – упрощенный неформальный язык, опускающий подробности синтаксиса, легко понятный человеку. Используется на ранних стадиях разработки программ, требует перевода на языки программирования.

Основные операторы псевдокода приведены в таблице

Название структуры	Псевдокод
присваивание, ввод, вывод	переменная = 0, ввод (переменная), вывод (переменная)
ветвление	<u>если</u> условие <u>то</u> (серия1 <u>иначе</u> серия 2)
цикл ПОКА	<u>пока</u> условие <u>нц</u> серия кц

Развитие языков программирования идет по пути сближения с псевдокодом.

2. Блок-схемы – графическое отображение псевдокода с помощью специальных блоков (прямоугольных, овальных и т.д.) и соединяющих стрелок.

3. Языки программирования.

Язык программирования — формальная знаковая система, предназначенная для записи компьютерных программ. Язык программирования определяет набор лексических, синтаксических и семантических правил, задающих внешний вид программы и действия, которые выполнит исполнитель (компьютер) под ее управлением.

## 5.3 Языки программирования.

Поскольку в качестве основного исполнителя алгоритмов подразумевается компьютер, то алгоритмы должны быть записаны на понятном ему языке. Такого рода языки наз. Я.П. Более точно, процессор понимает язык так наз. машинных кодов, однако, этот язык настолько неудобен для записи алгоритмов, что пришлось прибегнуть к созданию более удобных для человека языков, которые переводятся на язык машинных кодов специальными программами - компиляторами или трансляторами.

Языки программирования можно разделить на 3 класса по уровню: языки низкого уровня, языки высокого уровня и языки сверхвысокого уровня.

Языки низкого уровня: собственно машинный код и ассемблер. Команды маш. кода - это числа, указывающие процессору элементарные действия: поместить число в указанную ячейку памяти, прочитать число из указанной ячейки памяти, сложить два числа и т.д.

Ассемблер - те же самые маш. коды, но записанные в виде коротких слов (2-4 буквы).

Применение: драйверы устройств, антивирусы, небольшие системные программы.

Преимущества: компактность, быстрое действие. Недостатки: трудоемкость разработки, отсутствие переносимости (машинная зависимость).

Языки высокого уровня: компилируемые и интерпретируемые.

1. Компилируемые языки: С и его модификации, Паскаль, Фортран и т.д. К.Я. переводятся на язык маш. кодов компилятором и преобразуются в исполнимый модуль (в системе Windows имеет расширение .exe), который может быть запущен, как отдельная программа. Преимущества: быстрое действие, не требуют для выполнения дополнительных программ. Недостатки: трудно вносить изменения в код, возможна машинная зависимость.

2. Интерпретируемые языки: Бейсик, Перл, Питон. И.Я. переводятся на язык маш. кодов специальной программой - интерпретатором - каждый раз непосредственно во время выполнения. Преимущества: легко вносить изменения в код, высокая степень переносимости,

легко освоить программирование. Недостатки: медленные, требуют наличия интерпретатора. Деление на компилируемые и интерпретируемые языки условно. Существуют компиляторы для некоторых интерпретируемых языков, иногда возникает необходимость в интерпретации компилируемого языка.

3. Языки сверхвысокого уровня - скриптовые языки, нужны для автоматизации управления действиями некоторых сложных программ. Например, многие действия по редактированию текстов в MS Word можно автоматизировать, написав программу на встроенном языке программирования. Программы на Я.С.У. как правило называются скриптами, в некоторых случаях — макросами, они применимы только для какого-то одного пакета программ. Относятся к интерпретируемым.

Программа, переводящая программный код с одного языка на другой, называется транслятором.

Более подробно с Я.П. Можно ознакомиться на странице

[http://ru.wikipedia.org/wiki/Языки программирования](http://ru.wikipedia.org/wiki/Языки_программирования)

## **6. Локальные и глобальные сети ЭВМ. Защита информации в сетях**

### **6.1 Определение**

Интернет — всемирная система объединённых компьютерных сетей, построенная на использовании протокола IP и маршрутизации пакетов данных. Интернет образует глобальное информационное пространство, служит физической основой для Всемирной паутины (WWW, World Wide Web) и множества других систем (протоколов) передачи данных.

В настоящее время, когда слово «Интернет» употребляется в обиходе, чаще всего имеется в виду Всемирная паутина и доступная в ней информация, а не сама физическая сеть.

### **6.2 Сервисы**

\* электронная почта (E-mail), обеспечивающая возможность обмена сообщениями одного человека с одним или несколькими абонентами;

\* телеконференции, или группы новостей (Usenet), обеспечивающие возможность коллективного обмена сообщениями;

\* сервис FTP — система файловых архивов, обеспечивающая хранение и пересылку файлов различных типов;

\* сервис Telnet, предназначенный для управления удалёнными компьютерами в терминальном режиме;

\* World Wide Web (WWW, W3) — гипертекстовая (гипермедиа) система, предназначенная для интеграции различных сетевых

ресурсов в единое информационное пространство;

- \* сервис DNS, или система доменных имен, обеспечивающий возможность использования для адресации узлов сети мнемонических

имен вместо числовых адресов;

- \* сервис IRC, предназначенный для поддержки текстового общения в реальном времени (chat);

- \* Потокковое мультимедиа.

Наряду со стандартными сервисами существуют и нестандартные, представляющие собой оригинальную разработку той или иной компании. В качестве примера можно привести различные системы типа Instant Messenger (своеобразные интернет-пейджеры — ICQ, AOL, Demos on-line и т. п.), системы интернет-телефонии, трансляции радио и видео и т. д.

### 6.3 Услуги

Сейчас наиболее популярные услуги Интернета — это:

- \* Всемирная паутина
  - о Веб-форумы
  - о Блоги
  - о Вики-проекты (и, в частности, Википедия)
  - о Интернет-магазины
  - о Интернет-аукционы
  - о Социальные сети
- \* Электронная почта и списки рассылки
- \* Группы новостей (в основном, Usenet)
- \* Файлообменные сети
- \* Электронные платёжные системы
- \* Интернет-радио
- \* Интернет-телевидение
- \* IP-телефония
- \* Мессенджеры
- \* FTP-серверы
- \* IRC (реализовано также как веб-чаты)
- \* Поисковые системы
- \* Интернет-реклама
- \* Удалённые терминалы
- \* Удалённое управление
- \* Многопользовательские игры
- \* Web 2.0

### 6.4 Протоколы передачи информации

Протокол передачи информации - набор правил, позволяющий осуществлять соединение и обмен данными между двумя и более включёнными в сеть устройствами.

Протоколы делятся на протоколы низкого и высокого уровня.

1) протоколы низкого уровня образуют тесную связку, и обычно упоминаются совместно: TCP/IP. TCP (Transmission Control Protocol) следит за разбиением сообщения на небольшие кусочки (пакеты), целостностью пакетов при передаче и сборкой сообщения из пакетов. IP (Internet Protocol) следит за тем, чтобы пакеты были доставлены адресату. При этом разные пакеты одного сообщения могут доставляться адресату различными путями. В выборе правильного маршрута большую роль играют специальные устройства — маршрутизаторы.



2) протоколы высокого уровня надстраиваются над TCP/IP. Важно знать следующие:

а) HTTP (Hyper Text Transfer Protocol) – это протокол передачи гипертекста. Протокол HTTP используется при пересылке Web-страниц с одного компьютера на другой.

б) FTP (File Transfer Protocol)- это протокол передачи файлов со специального файлового сервера на компьютер пользователя.

в) POP (Post Office Protocol) и SMTP (Simple Mail Transfer Protocol) – протоколы передачи электронной почты (e-mail).

г) TELNET – это протокол удаленного доступа. TELNET дает возможность абоненту работать на любой ЭВМ сети Интернет, как на своей собственной.

д) XMPP – перспективный протокол, первоначально разработанный для передачи мгновенных сообщений Jabber (жаббер) – аналога ICQ, в настоящее время стремительно захватывает службы общения в Сети. Достаточно сказать, что на XMPP основана работа Facebook, Google Talk (и многих других сервисов Google), Яндекс, ВКонтакте и LiveJournal.

е) SSH (Secure Shell) - протокол для передачи данных в зашифрованном виде.

## Вопросы

1. Понятие информации и информатики.
2. Количество информации.
3. Кодирование информации.
4. Протоколы передачи информации.
5. Устройство компьютера.
6. Устройства хранения, передачи, преобразования, ввода и вывода информации.
7. Основные алгоритмические структуры (найти сам-но в сети).
8. Элементы графического окна программы: панели, области, настройка внешнего вида (на примере текстового или табличного редактора).
9. Форматы графических файлов. Программы для их обработки.
10. Основные типы компьютерной графики, их достоинства и недостатки. Цветовые модели.
11. Возможности растрового графического редактора на примере GIMP.
12. Электронные таблицы: назначение, возможности, приемы работы.
13. Основные службы Интернет (самостоятельно).
14. Работа с командной строкой (на примере ком. строки Linux)