

# Лекция № 2

## **Кодирование данных и структуры данных**

# План лекции

2.1. Кодирование данных.

2.1.1. Кодирование данных. Примеры

2.1.2. Кодирование данных двоичным кодом

2.1.3. Кодирование текстовых данных

2.1.4. Кодирование графических данных

2.1.4.1. Кодирование растровых данных

2.1.4.2. Кодирование векторных данных

2.1.5. Кодирование звуковой информации

2.2. Основные структуры данных

2.2.1. Линейные структуры

2.2.2. Табличные структуры данных

2.2.3. Иерархические структуры данных

2.2.4. Сетевые структуры данных

## 2.1. Кодирование данных.

Для автоматизации работы с данными, относящимися к различным типам, необходимо унифицировать их форму представления — для этого обычно используется **приём кодирования**.

**Кодирование** - выражение данных одного типа через данные другого типа.

**Естественные человеческие языки** — системы кодирования понятий для выражения мыслей посредством речи.

**Азбуки** - системы кодирования компонентов языка с помощью графических символов.

**Система записи математических выражений** — математическое кодирование.

# 2.1.1. Кодирование данных. Примеры

C	O	M	P	U	T	E	R	
43	4F	4D	50	55	54	45	52	Код ASCII
· · · · ·	· · · · ·	· · · · ·	· · · · ·	· · · · ·	· · · · ·	· · · · ·	· · · · ·	Код Морзе
● ●	● ● ●	● ● ●	● ● ● ●	● ● ●	● ● ● ●	● ● ●	● ● ● ●	Код Брайля
								Код морской сигнальный

Рис. 1.2. Примеры различных систем кодирования

## 2.1.2. Кодирование данных ДВОИЧНЫМ КОДОМ

- Система кодирования в вычислительной технике — **двоичное кодирование** - основана на представлении данных последовательностью всего двух знаков: 0 и 1.
- Эти знаки называются двоичными цифрами, по английски — **binary digit (bit)**.
- Одним битом могут быть выражены два понятия: 0 или 1 (да или нет, черное или белое, истина или ложь и т. п.)

## 2.1.2. Кодирование данных ДВОИЧНЫМ КОДОМ

Целые числа кодируются двоичным кодом достаточно просто — достаточно взять целое число и делить его пополам до тех пор, пока в остатке не образуется ноль или единица. Совокупность остатков от каждого деления, записанная справа налево вместе с последним остатком, и образует двоичный аналог десятичного числа.

$$19 : 2 = 9 + 1$$

$$9 : 2 = 4 + 1$$

$$4 : 2 = 2 + 0$$

$$2 : 2 = 1$$

Таким образом, **19 = 1011**.

# 2.1.3. Кодирование текстовых данных

Каждому символу алфавита сопоставляется определенное целое число

Таблица 1.1. Базовая таблица кодировки ASCII

32 пробел	48 0	64 @	80 P	96 `	112 p
33 !	49 1	65 A	81 Q	97 a	113 q
34 "	50 2	66 B	82 R	98 b	114 r
35 #	51 3	67 C	83 S	99 c	115 s
36 \$	52 4	68 D	84 T	100 d	116 t
37 %	53 5	69 E	85 U	101 e	117 u
38 &	54 6	70 F	86 V	102 f	118 v
39 ' .	55 7	71 G	87 W	103 g	119 w
40 (	56 8	72 H	88 X	104 h	120 x
41 )	57 9	73 I	89 Y	105 i	121 y
42 *	58 :	74 J	90 Z	106 j	122 z
43 +	59 ;	75 K	91 [	107 k	123 {
44 ,	60 <	76 L	92 \	108 l	124
45 -	61 =	77 M	93 ]	109 m	125 }
46 .	62 >	78 N	94 ^	110 n	126 ~
47 /	63 ?	79 O	95 _	111 o	127

## 2.1.4. Кодирование графических данных

Существует **два способа кодирования  
графических данных:**

- **Растровый**
- **Векторный**

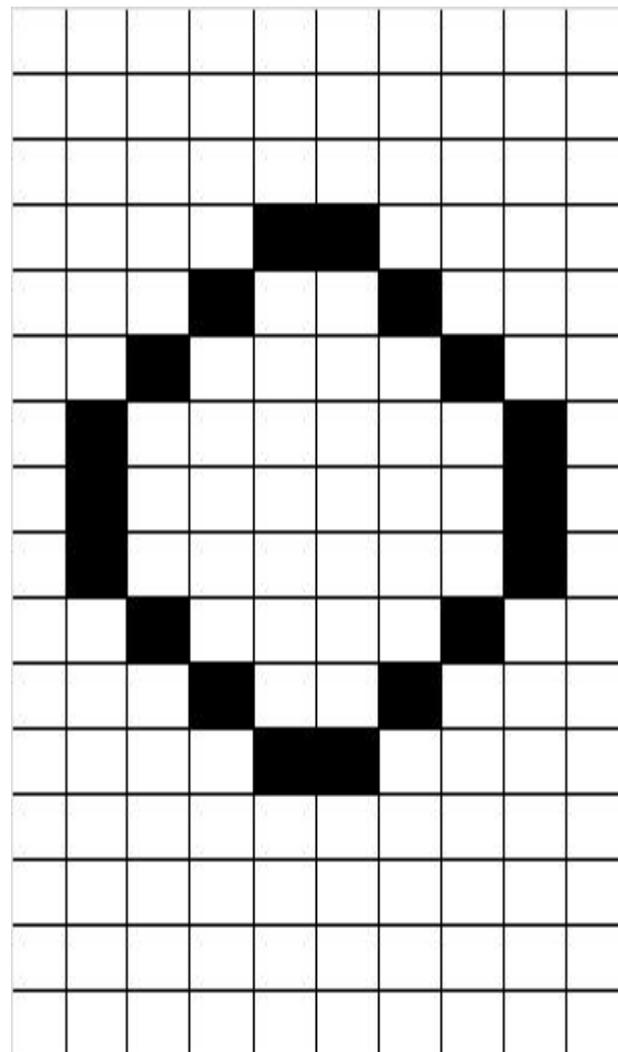
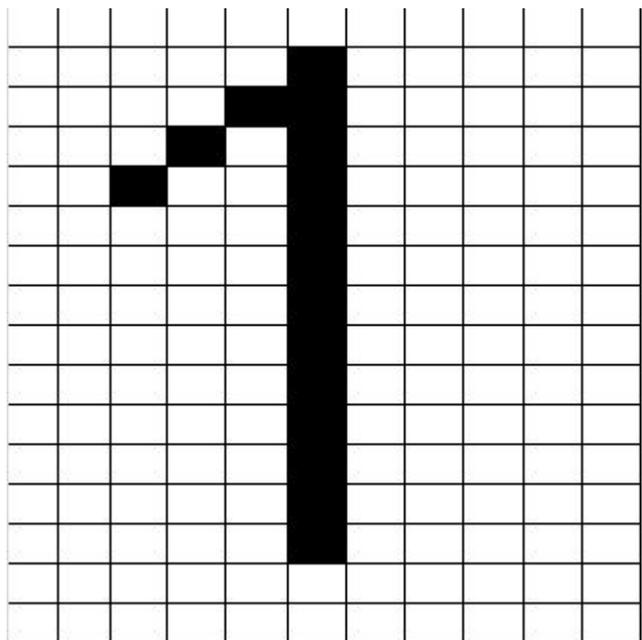
# 2.1.4.1. Кодирование растровых графических данных

**Растр** или растровый массив (**bitmap**), представляет совокупность битов, расположенных на сетчатом поле-канве.

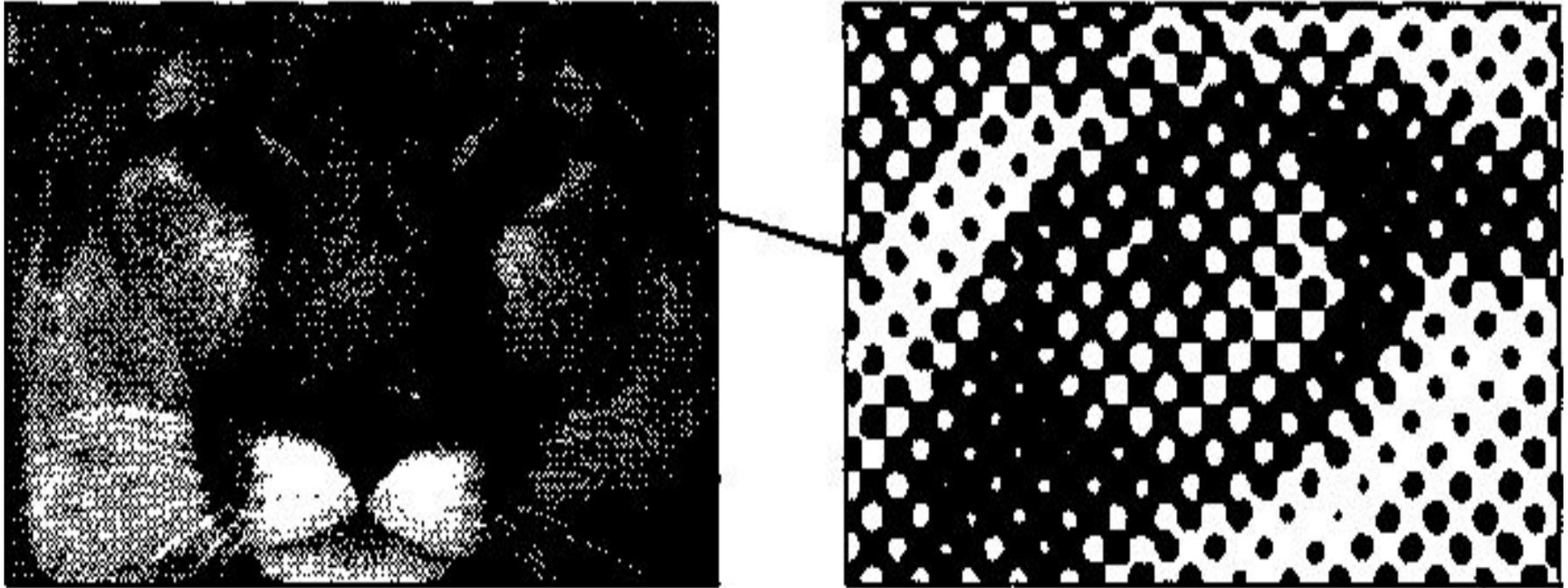
Бит может быть **включен** (единичное состояние) или **выключен** (нулевое состояние).

Состояние битов используется для представления черного или белого цветов, так что, соединив на канве несколько битов, можно создать изображение из черных и белых точек.

# 2.1.4.1. Кодирование растровых графических данных



# Пример кодирования черно-белых графических данных



На практике черно-белые рисунки представляются в виде комбинации точек с 256 градациями серого цвета. Для кодирования яркости любой точки достаточно восьмиразрядного двоичного числа.

# 2.1.4.1. Кодирование растровых графических данных

**Пиксель** — отдельный элемент растрового изображения;

Цвет каждого пикселя растрового изображения — черный, белый, серый или любой из спектра — запоминается с помощью комбинации битов. Чем больше битов используется для этого, тем большее количество оттенков цветов для каждого пикселя можно получить.

Число битов, используемых компьютером для хранения информации о каждом пикселе, называется **битовой глубиной** или **глубиной цвета**.

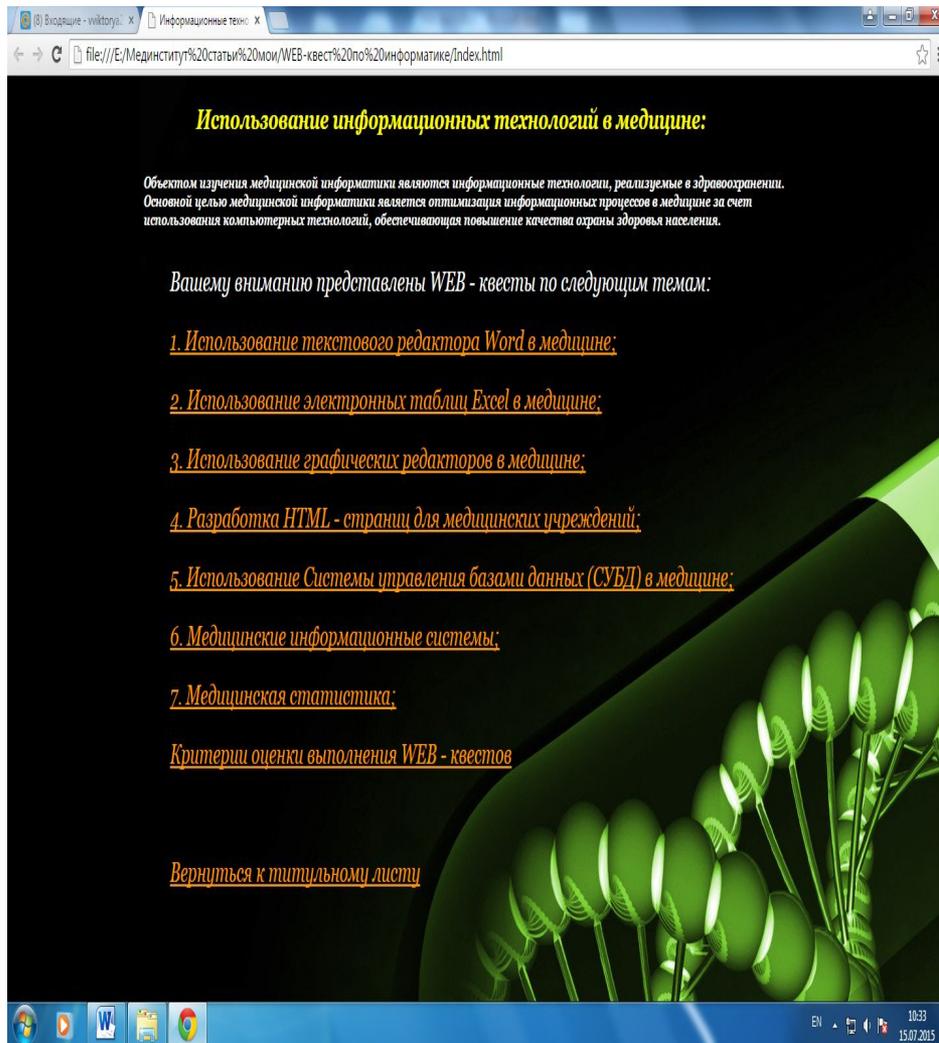
## 2.1.4.1. Кодирование растровых графических данных

Для отображения большего количества цветов используется больше битов информации.

Число возможных и доступных цветов или градаций серого цвета каждого пикселя равно двум в степени, равной количеству битов, отводимых для каждого пикселя.

24 бита обеспечивают более 16 миллионов цветов. О 24-битовых изображениях говорят как об изображениях с естественными цветами, так как такого количества цветов достаточно, чтобы отобразить всевозможные цвета, которые способен различать человеческий глаз.

# Пример кодирования цветных графических данных



Использование информационных технологий в медицине:

Объектом изучения медицинской информатики являются информационные технологии, реализуемые в здравоохранении. Основной целью медицинской информатики является оптимизация информационных процессов в медицине за счет использования компьютерных технологий, обеспечивающая повышение качества охраны здоровья населения.

Вашему вниманию представлены WEB - квесты по следующим темам:

- [1. Использование текстового редактора Word в медицине;](#)
- [2. Использование электронных таблиц Excel в медицине;](#)
- [3. Использование графических редакторов в медицине;](#)
- [4. Разработка HTML - страниц для медицинских учреждений;](#)
- [5. Использование Системы управления базами данных \(СУБД\) в медицине;](#)
- [6. Медицинские информационные системы;](#)
- [7. Медицинская статистика;](#)

[Критерии оценки выполнения WEB - квестов](#)

[Вернуться к титульному листу](#)



## 2.1.4.1. Кодирование векторных графических данных

**Векторное представление графических данных** - описание изображения в виде линий и фигур с использованием математических формул.

Векторное представление возможно с закрашенными областями, заполняемыми сплошным или градиентным цветом.

Хотя это может показаться более сложным, чем использование растровых массивов, но для многих видов изображений использование математических описаний является более простым способом.

# 2.1.4.1. Кодирование векторных графических данных

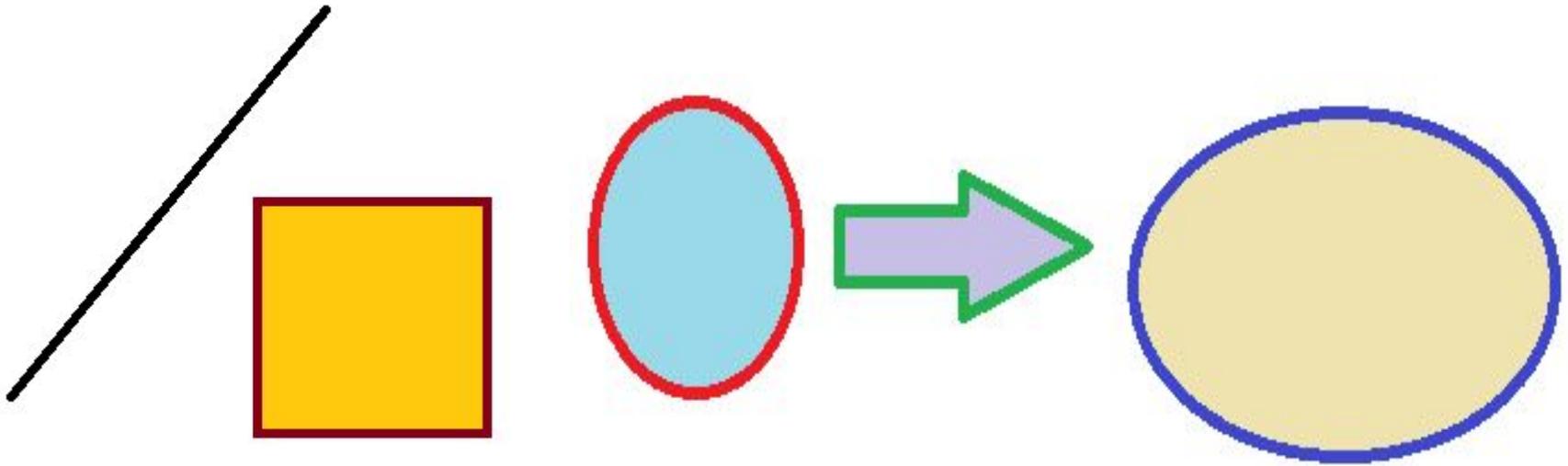
В векторной графике для описания объектов используются комбинации компьютерных команд и математических формул для описания объектов.

Это позволяет различным устройствам компьютера, таким как монитор и принтер, при рисовании этих объектов вычислять, где необходимо помещать реальные точки.

Векторную графику часто называют объектно-ориентированной или чертежной графикой.

Имеется ряд простейших объектов, или **примитивов**, например: эллипс, прямоугольник, линия. Эти примитивы и их комбинации используются для создания более сложных изображений.

## 2.1.4.1. Кодирование векторных графических данных



# 2.1.5. Кодирование звуковой информации

- **Метод FM** (Frequency Modulation) - любой сложный звук можно разложить на последовательность простейших гармонических сигналов разных частот, каждый из которых представляет собой правильную синусоиду, т.е. может быть описан числовым кодом. Их кодирование и декодирование выполняют специальные устройства АЦП-ЦАП.
- **Метод таблично-волнового (Wave-Table) синтеза** использует образцы звуков множества различных инструментов, которые хранятся в заранее подготовленных таблицах. В технике такие образцы называют сэмплами.

# 2.2. Основные структуры данных

## 2.2.1. Линейные структуры

- **Линейные структуры** — это упорядоченные структуры, в которых адрес элемента однозначно определяется его номером.
- **списки данных**

№	Фамилия И.О.
1	Афанасьева И.М.
2	Бобров В.В.
3	Воробьева О.С.
...	...
27	Якушкин А.С.

- Афанасьева И.М.#Бобров В.В.#Воробьева О.С.#...#Якушкин А.С.
- 
- **векторы данных** - все элементы списка имеют равную длину

## 2.2.2. Табличные структуры данных

- **Табличные структуры** - это упорядоченные структуры, в которых адрес элемента определяется номером строки и номером столбца, на пересечении которых находится ячейка, содержащая искомый элемент
- **таблицы данных**

№	ФИО	Дата рождения	температура
1	Егорова А.А.	05.03.1995	36,6
2	Жуков В.Д.	03.08.1993	37,8
3	Иванов И.И.	11.09.1989	38,2
4	Петров А.П	12.07.1993	39,0
5	Чернова М.Г.	14.01.1991	37,5

- **матрицы данных** — все элементы таблицы имеют одинаковую длину

## 2.2.2. Табличные структуры данных. Пример

ФИО доктора	Специальность	Кабинет
Иванова А.Г.	терапевт	2
Зимний Д.И.	окулист	3
Серегина К.И.	терапевт	4
Петрова А.Н.	хирург	5
Кривоносова О.В.	невропатолог	6
Шевченко Ф.Р.	кардиолог	7
Державина О.Д.	эндокринолог	8

## 2.2.2. Табличные структуры данных. Пример

**Количество введения анестетика в детской стоматологии при проводниковой анестезии.**

**Лидокаин с вазоконстриктором и без вазоконстриктора.**

**Максимальная доза – 4,4 мг/кг 2% р-р**

Вес	мг	мл	Ампул (в 1 ампуле–2мл(40мг))
10	44	2,3	1
15	66	3,3	1,5
20	88	4,4	2
25	110	5,5	2,5
30	132	6,6	3
35	154	7,7	3,5
40	176	8,8	4

## 2.2.3. Иерархические структуры данных

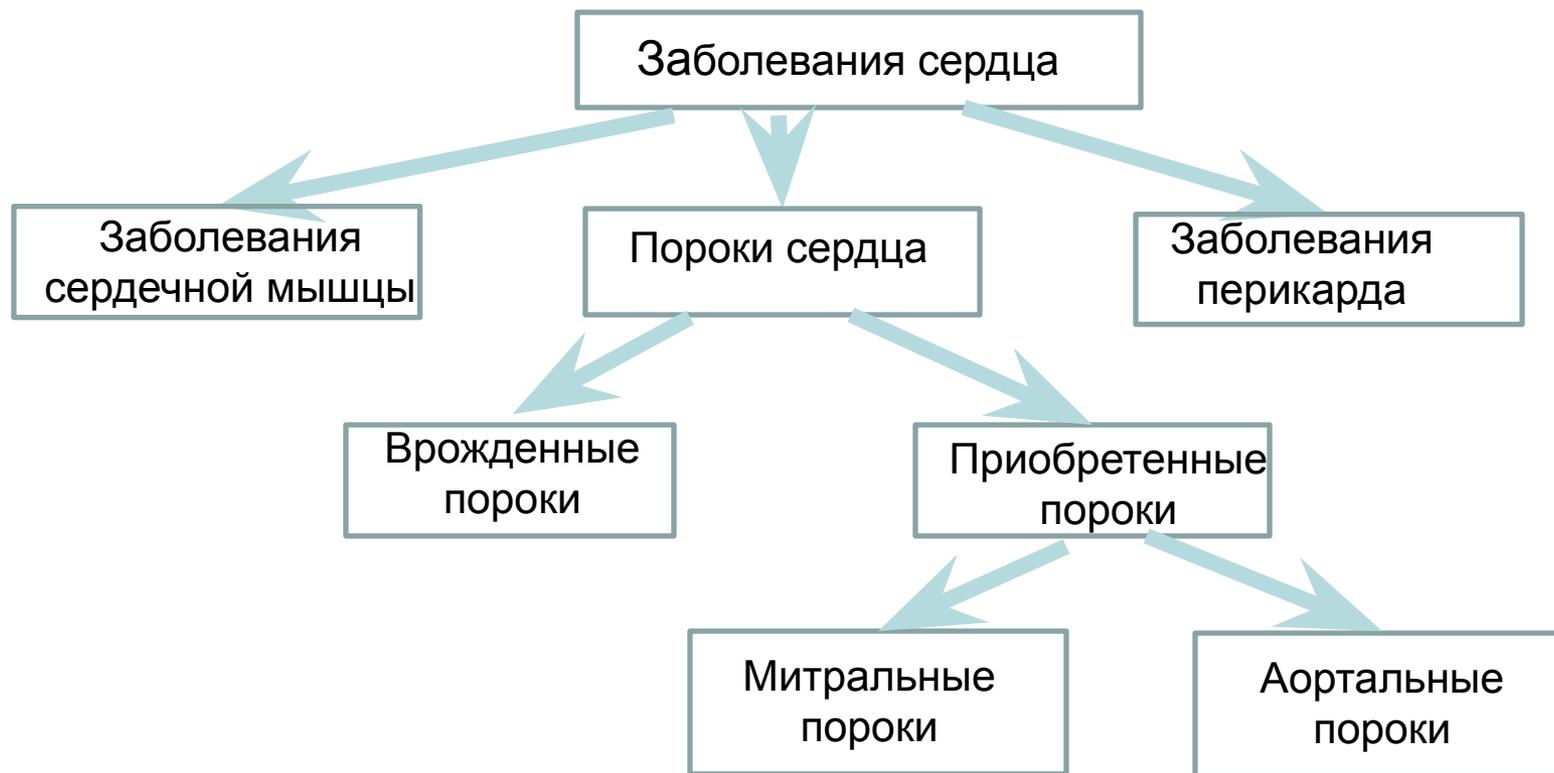
**Нерегулярные данные**, которые трудно представляются в виде списка или таблицы, могут быть представлены в **иерархической структуре**.

**Иерархическая структура данных** – это упорядоченная структура, в которой адрес каждого элемента определяется путем (маршрутом доступа), идущим от вершины структуры к данному элементу. В иерархической структуре элементы распределены по уровням.

Каждый элемент более высокого уровня может состоять из элементов нижнего уровня, а элемент нижнего уровня может входить в состав **только одного** элемента более высокого уровня.

## 2.2.3. Иерархическая структура данных. Пример

**Иерархические структуры данных** – представление нерегулярных данных



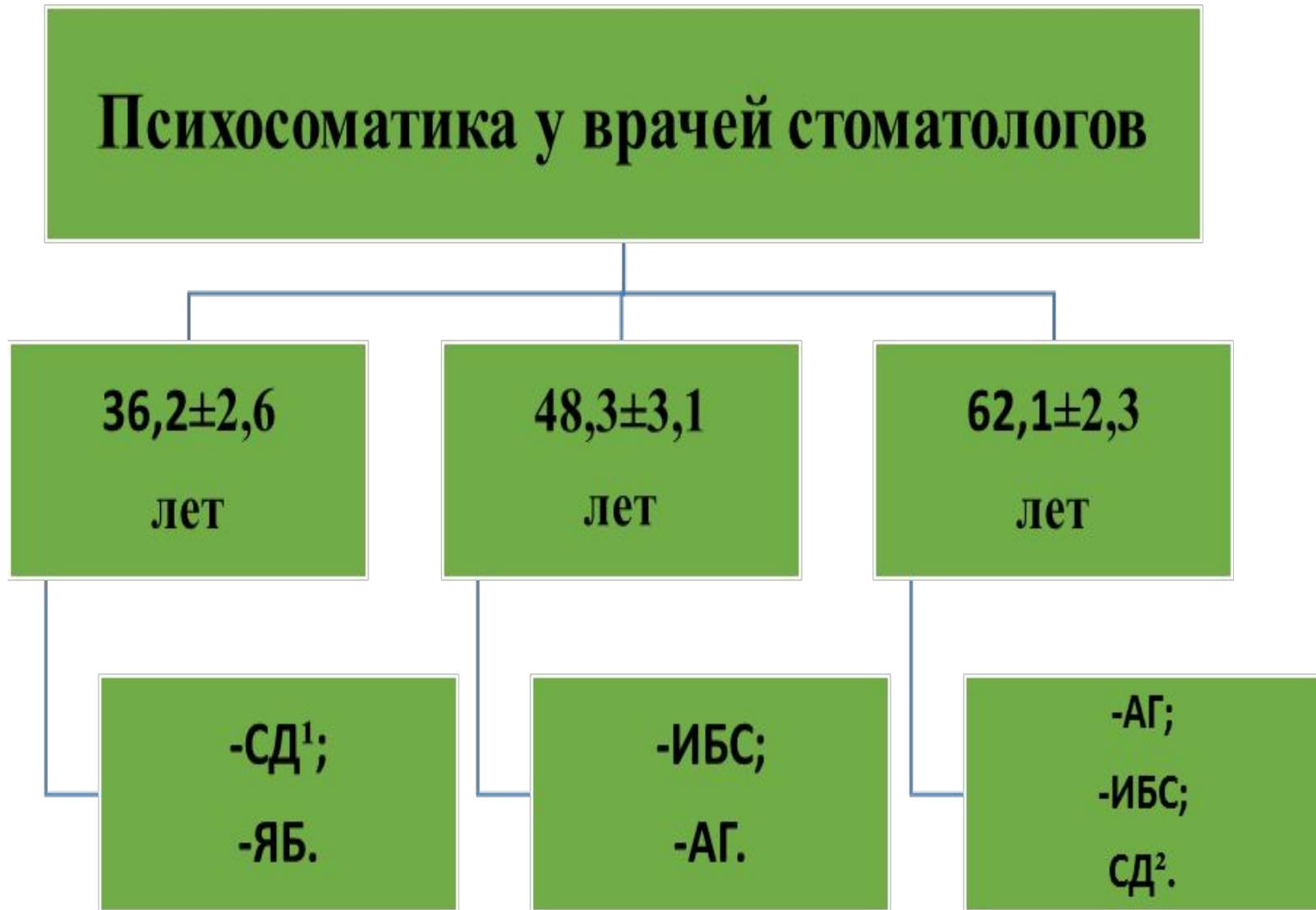
## 2.2.3. Иерархическая структура данных. Пример

### **ОТТИСК-**

это обратное (негативное) отображение поверхности твердых и мягких тканей, расположенных на протезном ложе и его границах, полученное с помощью специальных материалов.



## 2.2.3. Иерархическая структура данных. Пример



## 2.2.3. Иерархическая структура данных. Пример

Схема иерархической структуры данных в медицинских информационных системах



## 2.2.4. Сетевые структуры данных

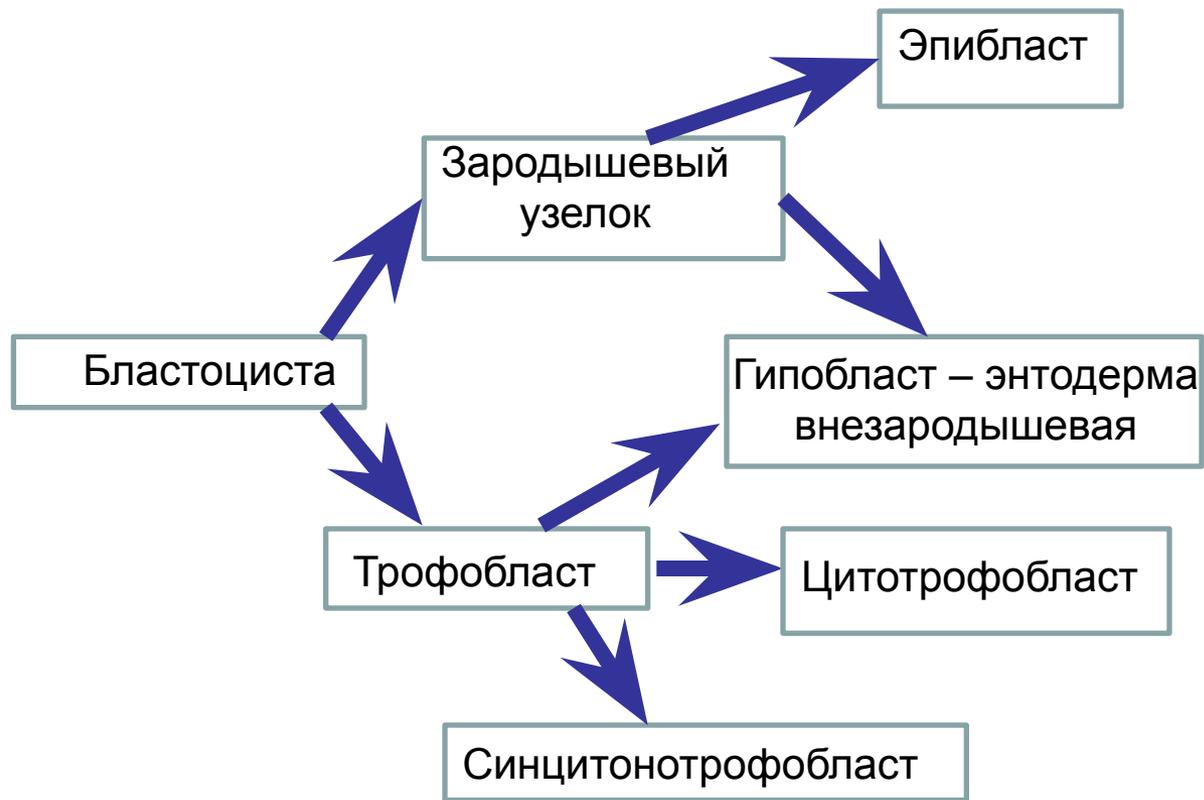
**Сетевые структуры** - каждый порожденный элемент может иметь более одного порождающего элемента. Сетевая модель данных отличается от иерархической тем, что каждый элемент сетевой структуры данных может быть связан с любым другим элементом.

Примером сложной сетевой структуры может служить структура базы данных, содержащей сведения о студентах, участвующих в научно-исследовательских работах (НИРС). Возможно участие одного студента в нескольких НИРС, а также участие нескольких студентов в разработке одной НИРС.

Студент (номер зачетной книжки, фамилия, группа)



## 2.2.4. Сетевая структура данных. Пример Дифференцировка зародышевых листков млекопитающих



# Единицы измерения данных

- 1 байт = 8 бит
- 1 полуслово = 2 байта
- 1 слово = 4 байта
- 1 Кбайт = 1024 байт =  $2^{10}$  байт
- 1 Мбайт = 1024 Кбайт =  $2^{20}$  байт
- 1 Гбайт = 1024 Мбайт =  $2^{30}$  байт
- 1 Тбайт = 1024 Гбайт =  $2^{40}$  байт
- 1 страница неформатированного машинописного текста составляет около двух килобайт
- 1 фотография среднего качества составляет около одного Мегабайта

**На сегодня все...**

**Благодарю  
за внимание !!!**