

Лекция 10. ОБРАБОТКА ГРАФИЧЕСКОЙ ИНФОРМАЦИИ НА ЭВМ, СИСТЕМЫ МАШИННОЙ ГРАФИКИ.

Как и любая другая информация в ПЭВМ, графические изображения хранятся, обрабатываются и передаются по линиям связи в закодированном виде - т.е. в виде большого числа бит- нулей и единиц. Существует большое число разнообразных программ- **графических редакторов** , работающих с графическими изображениями. В них используются самые разные графические форматы- т.е. способы кодирования графической информации. Расширения имен файлов, содержащих изображение, указывают на то, какой формат в нем использован, а значит какими программами его можно просмотреть, изменить (отредактировать), распечатать.



Рис.1. Виды компьютерной графики

1. Растровая графика

В **растровой графике** все изображение разбивается на маленькие "точки" -- пиксели, оттенок каждого из которых обозначают двоичным кодом. В простейшем случае картинка состоит только из черных и белых точек. Такую точку можно обозначить одним двоичным символом, допустим, ноль будет обозначать черный цвет, а единица -- белый. Значит, для хранения такого изображения нужно будет столько бит памяти, из скольких пикселей оно состоит. Например, первые компьютеры "Макинтош" имели разрешение экрана 512 x 342 пиксель. То есть, все изображение состояло из $512 \cdot 342 = 175104$ элементов. Для хранения каждого использовался 1 бит памяти.

Следовательно все изображение занимало 175104 бит = 21888 байт -- (приблизительно 21 Кбайт).

А если изображение цветное и состоит не только из черных и белых точек? Увеличим количество разрядов в коде: пусть каждый пиксель обозначается двумя битами. Сколько разных кодов получится? "00", "01", "10" и "11" -- четыре. Мало? Добавим еще бит: "000", "001", "010", "011", "100", "101", "110", "111" -- можно закодировать 8 цветов. Как видите, увеличение длины кода на один символ удваивает количество различных оттенков, которые можно обозначить.

бит										1
									0	
цвето										1
в			6	2	4	28	56	12	024	

Для того чтобы человеческий глаз не замечал мерцания монитора лучи должны пробегать весь экран несколько десятков раз в секунду²⁷. Поэтому картинка, которую показывает нам компьютер, помещается в специальную оперативную память -- **видеопамять**, из которой информация и считывается с необходимой частотой.

Надо **рассчитать минимальный объем видеопамяти** для компьютера, который при разрешении 640x480 пиксел отображает 256 оттенков. Изображение в этом случае состоит из $640 \cdot 480 = 307200$ элементов. Чтобы можно было обозначить 256 цветов, код каждого из них должен содержать (посмотрите в таблицу) 8 бит. Значит, каждый пиксел займет в памяти 8 бит = 1 байт, а весь экран -- 307200 байт = 300 Кбайт. Современные компьютеры могут работать с гораздо большим разрешением и обрабатывать изображения, содержащие тысячи²⁸ и даже миллионы²⁹ оттенков цвета, кроме того, в видеопамяти может храниться несколько кадров одновременно, поэтому объем её может достигать нескольких мегабайт.

Для работы с графической информацией существуют специальные программы, которые так и называются -- **графические редакторы**. Все они позволяют нарисовать нужное изображение, изменить его, сохранить рисунок во внешней памяти, а также напечатать его. Однако, закодировать картинку для обработки можно по-разному. Вы уже знаете о растровом представлении изображения. В этом случае наименьший элемент, который можно изменить, -- это отдельная точка растра (пиксел). Конечно, можно работать не только с отдельными точками, но и любыми произвольными фрагментами рисунка, имитировать рисование различными реальными инструментами художника. Редакторы, использующие такое представление информации, называются **растровыми**.

Основным недостатком растровой графики является большой объем памяти, требуемый для хранения изображения. Это объясняется тем, что нужно запомнить цвет каждого пиксела, общее число которых может быть очень большим. Например, одна фотография среднего размера в памяти компьютера занимает несколько Мегабайт, т.е. столько же, сколько несколько сотен (а то и тысяч) страниц текста.

2. Векторная графика .

При использовании векторной графики в памяти ЭВМ сохраняется математическое описание каждого графического примитива-геометрического объекта (например, отрезка, окружности, прямоугольника и т.п.), из которых формируется изображение.

В частности, для отрисовки окружности достаточно запомнить положение ее центра, радиус, толщину и цвет линии. По этим данным соответствующие программы построят нужную фигуру на экране дисплея. Понятно, что такое описание изображения требует намного меньше памяти (в 10 - 1000 раз) чем в растровой графике, поскольку обходится без запоминания цвета каждой точки рисунка.

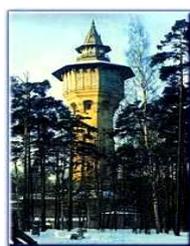
Основным недостатком векторной графики является невозможность работы с высококачественными художественными изображениями,

фотографиями и фильмами. Природа избегает прямых линий, правильных окружностей и дуг. К сожалению, именно с их помощью (поскольку эти фигуры можно описать средствами математики, точнее- аналитической геометрии) и формируется изображение при использовании векторной графики.

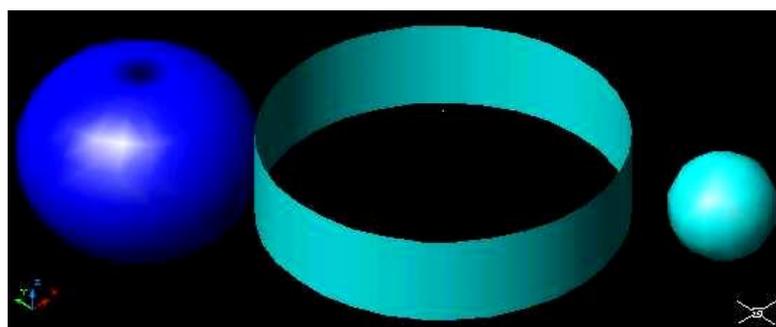
Основной сферой применения векторной графики является отрисовка чертежей, схем, диаграмм и т.п.

Как отличить векторную графику от растровой? Если Вы видите на экране фотографию или рисунок с близким к естественному изображением, с большим числом цветов и оттенков, то, скорее всего, Вы имеете дело с растровой графикой. Если чертеж, диаграмму, простой стилизованный рисунок,- с векторной. Если программа позволяет стирать, копировать или перемещать целые фрагменты (площади) изображения, то это растровая графика. Если удалить, скопировать, переместить можно только какие-то определенные фигуры или их части, то это графика векторная.

Пример изображения, созданного с использованием **растровой** графики:



Пример изображения, созданного с использованием **векторной** графики:



Файлы ***.bmp , *.psx , *.gif , *.msp , *.img** и др. соответствуют форматам **растрового** типа, Файлы ***.dwg , *.dxf , *.pic** и др. - файлы **векторного** типа.

Иногда **растровые** изображения могут входить в состав **векторных** как **отдельные графические примитивы**.