Компьютерная графика – область деятельности, в которой компьютеры используются в качестве инструмента для создания изображений, а также для обработки визуальной информации, полученной из реального мира.

Двумерная графика – это изображение на плоскости, имеющее длину и ширину. Двумерная компьютерная графика классифицируется по типу представления графической информации и следующими из него алгоритмами обработки изображений. Обычно компьютерную графику разделяют на *векторную* и *растровую,* хотя обособляют еще и *фрактальную графику.*

*Примитив* (графический примитив) – простейшая геометрическая фигура.

Трёхмерная графика — раздел компьютерной графики, посвящённый методам создания изображений или видео путём моделирования объёмных объектов в трёхмерном пространстве.

Текстура — это двумерный рисунок, который накладывается на 3D модель.

Язык программирования — формальный язык, предназначенный для записи компьютерных программ.

Можно выделить три принципиально разных способа реализации языков программирования: компиляция, интерпретация и встраивание.

Компиляция означает, что исходный код программы сперва преобразуется в целевой (машинный) код специальной программой, называемой компилятором.

Интерпретация же означает, что исходный код выполняется непосредственно, команда за командой.

Встраивание языка можно философски рассматривать как «реализацию без трансляции» — в том смысле, что такой язык является синтаксическим и семантическим подмножеством некоего другого языка, без которого он не существует.

Почему pascal, java, javascript?

Pascal:

1 Язык созданый ученым кибернетиком специально для обучения алгоритмическому программированию;

2 Реализует все базовые алгоритмические конструкции;

3 Строго типизированный. Ошибки многих видов выявляются на этапе компиляции;

4 Упрощает представление и обработку сложных структур данных;

Java:

1 полностью реализует концепцию объектно ориентированного программирования;

2 Использование виртуальной машины обеспечивает переносимость кода на различные платформы без перекомпиляции;

3 Эффективно реализованная многопоточность;

4 Высокая производительность обеспечивается глубокой оптимизацией виртуальной машины;

5 Очень высокая безопасность за счет контролируемого исполнения кода

6 Автоматическое управление памяти(сборка мусора)

7 Большое количество готовых компонентов и средств разработки

JavaScript:

2Обеспечивает высокую производительность за счет однопоточности и принципа обратных вызовов

1 Клиентские приложения исполняются всеми браузерами на всех платформах, не требует компиляции.

3 Легкость распространения кода

4 Эффективно реализованные, хорошо документированные, стандартизированные средства разработки, значительно повышающие производительность труда

5 Автоматическое управление памяти(сборка мусора)

Графические примитивы представляют собой процедуры, осуществляющие рисование в графическом окне. Рисование осуществляется текущим пером (линии), текущей кистью (заливка замкнутых областей) и текущим шрифтом (вывод строк).

Среда разработки Lazarus

1. бесплатно, минимум ресурсов;
2. максимально просто установить и использовать;
3. ориентрована на обучение;

SWING+GUI

В Java есть 2 основных пакета для создания графических интерфейсов (Graphics User Interface). Это Abstract Windows Toolkit (AWT) и Swing. AWT использует виджеты операционной системы, поэтому эта библиотека немного быстрее.

AWT была первой попыткой Sun создать графический интерфейс для Java. Они пошли легким путем и просто сделали прослойку на Java, которая вызывает методы из библиотек, написанных на С. Библиотечные методы создают и используют графические компоненты операционной среды. С одной стороны, это хорошо, так как программа на Java похожа на остальные программы в рамках данной ОС. Но с другой стороны, нет никакой гарантии, что различия в размерах компонентов и шрифтах не испортят внешний вид программы при запуске ее на другой платформе. Кроме того, чтобы обеспечить мультиплатформенность, пришлось унифицировать интерфейсы вызовов компонентов, из-за чего их функциональность получилась немного урезанной. Да и набор компонентов получился довольно небольшой. К примеру, в AWT нет таблиц, а в кнопках не поддерживается отображение иконок.

Вслед за AWT Sun разработала набор графических компонентов под названием Swing. Компоненты Swing полностью написаны на Java. Для отрисовки используется 2D, что принесло с собой сразу несколько преимуществ. Набор стандартных компонентов значительно превосходит AWT по разнообразию и функциональности. Стало легко создавать новые компоненты, наследуясь от существующих и рисуя все, что душе угодно. Стала возможной поддержка различных стилей и скинов. Вместе с тем скорость работы первых версий Swing оставляла желать лучшего. Некорректно написанная программа и вовсе могла повесить Windows намертво.

Тем не менее, благодаря простоте использования, богатой документации и гибкости компонентов Swing стал, пожалуй, самым популярным графическим фреймворком в Java. На его базе появилось много расширений, таких как SwingX, JGoodies, которые значительно упрощают создание сложных пользовательских интерфейсов. Практически все популярные среды программирования Java включают графические редакторы для Swing-форм. Поэтому разобраться и начать использовать Swing не составит особого труда.