**1.1 Введение**

*Android* - *операционная система* для мобильных устройств: смартфонов, планшетных компьютеров, КПК. В настоящее время именно *Android*является самой широко используемой операционной системой для мобильных устройств. Подтверждение этого факта можно найти в таблице, составленной *по* данным аналитической компании Gartner.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Таблица 1.1. Мировые продажи смартфонов конечным пользователям, распределение по ОС | | | | |
| **Операционная система** | **Продано (тыс.ед.) III кв. 2013** | **Доля рынка (%) III кв. 2013** | **Продано (тыс.ед.) III кв. 2012** | **Доля рынка (%) III кв. 2012** |
| **Android** | 205022,7 | 81,9 | 124552,3 | 72,6 |
| **iOS** | 30330,0 | 12,1 | 24620,3 | 14,3 |
| **Microsoft** | 8912,3 | 3,6 | 3993,6 | 2,3 |
| **BlackBerry** | 4400,7 | 1,8 | 8946,8 | 5,2 |
| **Bada** | 633,3 | 0,3 | 4454,7 | 2,6 |
| **Symbian** | 457,5 | 0,2 | 4401,3 | 2,6 |
| **другие** | 475,2 | 0,2 | 683,7 | 0,4 |
| **Общее кол-во:** | **250231,7** | **100,0** | **171652,7** | **100,0** |
| Источник: Gartner (ноябрь 2013) | | | | |

Внимательное изучение таблицы позволяет увидеть подавляющую популярность смартфонов под управлением ОС *Android* в мире, доля таких устройств не первый год превышает половину от общего числа купленных смартфонов. Кроме всего прочего, эта популярность продолжает расти. Очевидно, что армия пользователей смартфонов под управлением *Android* будет искать дополнительные приложения для своих устройств, в связи с этим умение разрабатывать эти самые приложения может принести много пользы своему владельцу. Например, можно разрабатывать для себя полезные, интересные, занимательные (нужное подчеркнуть) приложения, а можно, разведав обстановку и осмотревшись, сделать разработку мобильных приложений своей профессиональной деятельностью, основной или дополнительной.

Курс "Разработка приложений для смартфонов на ОС *Android*" предоставляет возможность приобрести начальные навыки разработки мобильных приложений, если остановиться только на первой его части. Изучение полной версии курса позволит сделать серьезный шаг к тому, чтобы профессионально разрабатывать мобильные приложения и получать от этой деятельности не только моральное, но и материальное удовлетворение.

Данная лекция является первой для всего курса, призвана ввести читателя в курс дела. В первую *очередь* в ней рассматриваются вопросы становления и развития ОС *Android*. Для успешного программирования под *Android* необходимо понимать внутреннюю организацию и архитектуру этой платформы, а также полезно знать, какие инструменты и среды разработки можно использовать. Этим вопросам посвящена основная часть лекции. Кроме того, в лекции рассматриваются особенности запуска и отладки мобильных приложений.

**Немного истории**

Рассмотрим, как все начиналось. В 2003 году в Пало Альто, штат Калифорния Энди Рубин с единомышленниками (Рич Майнер, Ник Сирс и Крис Уайт) основали компанию Android Inc. Поначалу в компании занимались проектированием мобильных гаджетов, которые на основе геолокационных данных автоматически подстраивались под нужды пользователей.

В августе 2005 года Android Inc. стала дочерней компанией Google. Энди Рубин, Рич Майнер и Крис Уайт остались в Android Inc. и начали работать над операционной системой, базирующейся на ядре Linux. В Google задумали реализовать мощнейшую платформу, пригодную к использованию на тысячах различных моделей телефонов. В связи с этим был создан Open Handset Alliance (OHA) - консорциум, состоящий из более 80 компаний, направляющий свои усилия на разработку открытых стандартов для мобильных устройств. В состав OHA входят такие гиганты, как Google (организатор и идейный вдохновитель), HTC, Sony, Dell, Intel, Motorola, Qualcomm, Texas Instruments, Samsung Electronics, LG Electronics, T-Mobile, Sprint Corporation, NVIDIA и многие другие.

Первая версия Android была представлена 23 сентября 2008 года, версии было дано название Apple Pie (можно заметить созвучие с прямым конкурентом). Далее так повелось, что название каждой очередной версии представляет какой-либо десерт, при этом первые буквы наименований в порядке версий соответствуют буквам латинского алфавита по порядку. С развитием обновлений Android можно познакомиться в [таблице 1.2](http://www.intuit.ru/studies/courses/12643/1191/lecture/21980?page=1#table.1.2).

|  |  |
| --- | --- |
| Таблица 1.2. История обновлений ОС Android | |
| **Версия, логотип, дата выхода** | **Основные возможности** |
| Android 1.0  http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_a01.jpg  Apple Pie | |  | | --- | | Первый стабильный релиз, основан на ядре Linux 2.6.25. | | Поддерживается:   * файловая система FAT32, стек интернет-протоколов TCP/IP; * протоколы передачи данных: 802.11 b/g Wi-Fi,Bluetooth 2.0 EDR, GPRS, EDGE, UMTS, HSDPA; * фото и видео съемка, однако недостаточно опций для настройки разрешения камеры, баланса белого и др.; * сенсорные дисплеи и landscape режим отображения данных на экране, максимальная цветность дисплея - 16 бит (тип HVGA); * виджеты и ярлыки на рабочем столе (Home Screen), сменные обои; * регулярные телефонные функции, контроль вызова, конференц-связь, легкая интеграция с контактами; * полноценный web-браузер на движке WebKit, HTML, XHTML; * e-mail клиент, протоколы POP3, IMAP4, SMTP; * медиа проигрыватель, позволяющий управлять, импортировать, проигрывать медиа контент в различных форматах. | | Базовые приложения: | | будильник; калькулятор; календарь; камера; контакты; сообщения (в том числе MMS); настройки; голосовой набор. | | Минимальные системные требования для запуска и работы: | | архитектура ARM, 128 MB RAM, 256 MB ROM. | | Видео презентация: | | <http://www.youtube.com/watch?feature=player_embedded&v=1FJHYqE0RDg> | |
| Android 1.1  http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_a01.jpg   |  | | --- | | Banana Bread | | февраль 2009 | | (API level: 2) | | |  | | --- | | Нововведения: | | Исправлены проблемы: с будильником; со спящим режимом; с вызовом дисплея набора номера; в IMAP ошибки запроса пароля и др. | | Изменения API. | | Добавлены подробности и отзывы к картам. | | Добавлена поддержка вложений из MMS. | | Локализации: | | Английская US (en\_US) | | Немецкая (de) | | Подробности: | | <http://developer.android.com/about/versions/android-1.1.html> | |
| |  | | --- | | Android 1.5 | | Cupcake |   http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_a02.jpg   |  | | --- | | апрель 2009 | | (API level: 3) | | |  | | --- | | Нововведения: | | Поддержка экранной клавиатуры (портретный и книжный режимы); акселерометра; видеозапись и воспроизведение видео; приложение для работы с YouTube; стерео Bluetooth; функция копирования и вставки между приложениями (copy&paste). | | Локализации: | | добавились очень многие, в том числе и русская (ru\_RU). | | Система: | | новое Linux ядро (версия 2.6.27); автоматическая проверка и восстановление файловой системы на SD card; новое приложение для просмотра СТК меню оператора (SIM Application Toolkit 1.0). | | Изменения в пользовательском интерфейсе (UI): | | изменено большинство UI-элементов, добавлены новые виджеты; определение режима (книжный или портретный) работы программы; анимированное переключение между окнами. | | Подробности: | | <http://developer.android.com/about/versions/android-1.5.html> | |
| Android 1.6  http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_a03.jpg   |  | | --- | | Donut | | сентябрь 2009 | | (API level: 4) | | |  | | --- | | Нововведения: | | Система: | | новое ядро Linux (версия 2.6.29); поддержка сотового стандарта CDMA; поддержка разрешений дисплеев: QVGA и WVGA; обновленный медиа-движок OpenCore 2; движок синтеза речи (многоязыковой); Gesture Builder поддержка возможности (для разработчиков) создавать, сохранять, загружать и распознавать жесты, прикреплять к определенным действиям. | | Пользовательские возможности: | | строка быстрого поиска (прямо с рабочего стола): история и закладки в браузере, контакты и поиск в интернете; возможность подключаться к видам VPN: L2TP/IPSEC pre-shared key based VPN, L2TP/IPSEC certificate based VPN, L2TP only VPN, PPTP only VPN; ускорение работы камеры; индикатор работы батареи позволяет увидеть сколько энергии потребляют работающие программы и сервисы. | | Обновленный Android Market. | | Подробности: | | <http://developer.android.com/about/versions/android-1.6.html> | |
| |  | | --- | | Android 2.0, | | 2.0.1, 2.1 |   http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_a04.jpg   |  | | --- | | Eclair | | октябрь 2009 | | (API level: 5) | | (API level: 6) | | (API level: 7) | | |  | | --- | | Нововведения в 2.0: | | поддержка работы нескольких почтовых аккаунтов одновременно, возможность использования совместных папок (входящие, исходящие) для всех аккаунтов; быстрый способ работы с контактами Quick Contact; | | поиск по всем сохраненным SMS и MMS сообщениям, удаление старых после заданного срока; | | возможности камеры: вспышка, цифровой зум, сценические режимы, баланс белого, цветовые эффекты, макрофокусировка; | | улучшенное расположение виртуальных клавиш клавиатуры, поддержка комбинированных нажатий клавиш (технология мультитач), усовершенствованная функция автодополнения; | | поддержка HTML5, версии Bluetooth 2.1, новых профилей OPP и PBAP. | | Подробности: | | <http://developer.android.com/about/versions/android-2.0.html> | | Нововведения 2.0.1: | | подрелиз версии 2.0, включающий в себя незначительные изменения в функционале и по большей части bugfix-ом версии 2.0. | | Подробности: | | <http://developer.android.com/about/versions/android-2.0.1.html> | | Нововведения 2.1: | | основным новшеством, представляющим интерес для конечного пользователя, стало добавление анимированных (живых) обоев, остальные изменения в Framework API, представляют интерес для разработчиков. | | Подробности: | | <http://developer.android.com/about/versions/android-2.1.html> | |
| Android 2.2  http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_a05.jpg   |  | | --- | | Froyo | | май 2010 | | (API level: 8) | | |  | | --- | | Нововведения: | | рост производительности примерно в 3-5 раз за счет использования Dalvik Virtual Machine Just-in-Time компилятора; | | возможности установки приложений на SD-карту, переноса приложений из внутренней памяти на карту и обратно; | | возможность использовать смартфон в качестве точки доступа к интернету, в качестве модема для других устройств; | | поддержка Adobe Flash; | | V8 javascript существенно повысил скорость работы штатного браузера. | | Подробности: | | <http://developer.android.com/about/versions/android-2.2.html> | |
| |  | | --- | | Android 2.3, | | 2.3.3 |   http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_a06.jpg   |  | | --- | | Gingerbread | | декабрь 2010 | | (API level: 9) | | (API level: 10) | | |  | | --- | | До весны 2013 года самая массовая версия на рынке. | | Нововведения: | | новое ядро Linux 2.6.35; поддержка открытых мультимедийных стандартов (VP8 и WebM), форматов ACC/AMR, звуковых эффектов и эквалайзера, фронтальной камеры (интеграция с VOIP(SIP)); | | обновленный GUI: уменьшение времени доступа к функциям, повышение общей энергоэффективности системы; | | улучшение стандартной клавиатуры системы: поддержка словарей, технологии мультитач, упрощенное выделение и копирование текста; | | поддержка технологии NFC; расширение возможностей работы с датчиками положения телефона. | | Подробности: | | <http://developer.android.com/about/versions/android-2.3.html> | |
| Android 3.0-3.2  http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_a07.jpg   |  | | --- | | Honeycomb | | февраль 2011 | | (API level: 11) | | (API level: 12) | | (API level: 13) | | |  | | --- | | Специальная версия для работы на планшетах (MID, tablets). | | Нововведения 3.0: | | новое ядро Linux 2.6.36; поддержка файловой системы ext4, файловой системы FUSE для MTP устройств; поддержка режима USB-хост для работы с клавиатурой, мышью и USB-хабами; поддержка MTP/PTP; | | виртуальная машина Dalvik: поддержка и оптимизация SMP, множество улучшений JIT, улучшенный сборщик мусора; | | совершенно новый интерфейс с полноценной оптимизацией под устройства с большими экранами; поддержка виртуальных рабочих столов, каждый из которых может иметь свой набор виджетов и ярлыков; | | улучшенные и переработанные базовые приложения: Browser, e-mail и др. | | Подробности: | | <http://developer.android.com/about/versions/android-3.0.html> | | Нововведения 3.1: | | поддержка работы кардридера; | | усовершенствован GUI: доработан менеджер задач, позволяющий переключаться между множеством различных приложений (в 3.0 только 5 программ одновременно); возможность менять размер виджетов, как по горизонтали, так и по вертикали. | | Подробности: | | <http://developer.android.com/about/versions/android-3.1.html> | | Нововведения 3.2: | | расширен спектр поддерживаемых планшетов; возможность автоматического масштабирования приложений для отображения на более крупных экранах. | | Подробности: | | <http://developer.android.com/about/versions/android-3.2.html> | |
| |  | | --- | | Android 4.0, 4.0.3 | | Ice Cream Sandwich |   http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_a08.jpg   |  | | --- | | ноябрь 2011 | | (API level: 14) | | (API level: 15) | | |  | | --- | | Нововведения: | | поддержка и смартфонов, и планшетов; поддержка новых процессорных архитектур, помимо ARM поддержка Intel x86 и MIPS; | | возможность разблокировки экрана: при помощи функции определения лица; жестами: перетащить замочек из центра экрана на иконку приложения и оно запустится; | | многозадачность: кнопка Recent Apps позволяет мгновенно переходить от одной задачи к другой с помощью списка в системной панели; | | новые элементы управления передачей данных через сеть: в приложении Настройки можно увидеть общее использование данных по каждому типу сети, объем данных, используемых каждым работающим приложением; | | доступность Android 4.0 для слепых и слабовидящих пользователей, браузер поддерживает экранного чтеца, который воспроизводит все видимое активное содержимое на экране; | | AndroidBeam - удобное средство обмена между двумя NFC-устройствами; | | Wi-Fi Direct и Bluetooth HDP, HFP: возможность прямого подключения к соответствующим устройствам. | | Подробности: | | <http://developer.android.com/about/versions/android-4.0.html> | |
| Android 4.1-4.3  http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_a09.jpg   |  | | --- | | Jelly Bean | | июль 2012 | | (API level: 16) | | (API level: 17) | | (API level: 18) | | |  | | --- | | Нововведения 4.1: | | увеличена скорость прорисовки интерфейса, улучшен поиск, добавлено несколько полезных сервисов; | | улучшена работа со словарями, возможно использовать голосовой ввод без подключения к интернету; | | специальные возможности: возможность управления смартфоном с помощью жестов и голосовых подсказок, подключения устройств ввода, поддерживающих шрифт Брайля; | | существенно доработана функция передачи данных Beam; переработан поиск (вместо ссылок ответ на запрос); голосовой поиск; Google Now: нужная информация в нужное время. | | Подробности: | | <http://developer.android.com/about/versions/android-4.1.html> | | Нововведения 4.2: | | реализована поддержка нескольких пользователей (планшеты); поддержка wireless display: возможность трансляции видео и изображений на внешний экран; возможность отображения полезной информации в режиме сна, при подключении к док-станции или на зарядке; улучшена панель уведомлений. | | Подробности: | | <http://android.com.ua/android_42.html> | | Нововведения 4.3: | | ускорение работы системы; более точный набор на клавиатуре; скрытая возможность управления процессами программ (необходима активация); поддержка OpenGL/ES 3.0 (не на всех устройствах). | | Подробности: | | <http://developer.android.com/about/versions/android-4.3.html> | |
| Android 4.4  http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_a10.jpg   |  | | --- | | Kit Kat | | октябрь 2013 | | (API level: 19) | | |  | | --- | | Нововведения: | | многозадачность, оптимизация распределения ресурсов между приложениями; определитель номера работает не только с адресной книгой (например, Google maps); серьезная интеграция приложения Hangouts (отправка SMS, MMS, голосовые и видеовызовы); в состав вошел Quickoffice, интегрированный с Google Drive; поддержка принтеров, подключение через приложения поддерживающие печать (например, Google Cloud Print, HP ePrint); поддержка стандарта Wi-Fi Miracast, позволяющий вещать изображение на телевизор; возможность захвата экрана для записи видео. | | Подробности: | | <http://developer.android.com/about/versions/android-4.4.html> | |

**1.2 Устройство платформы Android**

Платформа *Android* объединяет операционную систему, построенную на основе ядра ОС Linux, промежуточное *программное обеспечение* и встроенные мобильные приложения. Разработка и развитие мобильной платформы *Android* выполняется в рамках проекта AOSP (*Android* *Open Source Project*) под управлением OHA (*Open* *Handset* Alliance), руководит всем процессом поисковый гигант Google.

*Android* поддерживает фоновое выполнение задач; предоставляет богатую библиотеку элементов пользовательского интерфейса; поддерживает 2D и *3D* графику, используя OpenGL стандарт; поддерживает *доступ* к файловой системе и встроенной базе данных SQLite.

С точки зрения архитектуры, система *Android* представляет собой полный программный *стек*, в котором можно выделить следующие уровни:

* **Базовый уровень (Linux Kernel)** - уровень абстракции между аппаратным уровнем и программным стеком;
* **Набор библиотек и среда исполнения (Libraries & Android Runtime)** обеспечивает важнейший базовый функционал для приложений, содержит виртуальную машину Dalvik и базовые библиотеки Java необходимые для запуска Android приложений;
* **Уровень каркаса приложений (Application Framework)** обеспечивает разработчикам доступ к API, предоставляемым компонентами системы уровня библиотек;
* **Уровень приложений (Applications)** - набор предустановленных базовых приложений.

Наглядное изображение архитектуры на [рисунке 1.1](http://www.intuit.ru/studies/courses/12643/1191/lecture/21980?page=2#image.1.1).

[](http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_01.jpg)

**Рис. 1.1.**Архитектура Android

Рассмотрим компоненты платформы более подробно.

В основании компонентной иерархии лежит *ядро* ОС Linux 2.6 (несколько урезанное), служит промежуточным уровнем между аппаратным и программным обеспечением, обеспечивает функционирование системы, предоставляет системные службы ядра: *управление памятью*, энергосистемой и процессами, обеспечение безопасности, работа с сетью и драйверами.

Уровнем выше располагается набор библиотек и среда исполнения. Библиотеки реализуют следующие функции:

* предоставляют реализованные алгоритмы для вышележащих уровней;
* обеспечивает поддержку файловых форматов;
* осуществляет кодирование и декодирование информации (например, мультимедийные кодеки);
* выполняет отрисовку графики и т.д.

Библиотеки реализованы на С/С++ и скомпилированы под конкретное *аппаратное обеспечение* устройства, вместе с которым они и поставляются производителем в предустановленном виде.

Рассмотрим некоторые библиотеки:

|  |  |
| --- | --- |
| **Surface Manager** | - композитный менеджер окон. Поступающие команды отрисовки собираются в закадровый буфер, где они накапливаются, составляя некую композицию, а потом выводятся на экран. Это позволяет системе создавать интересные бесшовные эффекты, прозрачность окон и плавные переходы. |
| **Media Framework** | - библиотеки, реализованные на базе PacketVideo OpenCORE. Используются для записи и воспроизведения аудио и видео контента, а также для вывода статических изображений. Поддерживаются форматы: MPEG4, H.264, MP3, AAC, AMR, JPG и PNG. |
| **SQLite** | - легковесная и производительная реляционная СУБД, используется в Android в качестве основного движка для работы с базами данных. |
| **3D библиотеки** | - используются для высокооптимизированной отрисовки 3D-графики, при возможности используют аппаратное ускорение. Библиотеки реализованы на основе API OpenGL|ES. OpenGL|ES (OpenGL for Embedded Systems) - подмножество графического программного интерфейса OpenGL, адаптированное для работы на встраиваемых системах. |
| **FreeType** | - библиотека для работы с битовыми картами, для растеризации шрифтов и осуществления операций над ними. |
| **LibWebCore** | - библиотеки браузерного движка WebKit, используемого также в известных браузерах Google Chrome и Apple Safari. |
| **SGL (Skia Graphics Engine)** | - открытый движок для работы с 2D-графикой. Графическая библиотека является продуктом Google и часто используется в других программах. |
| **SSL** | - библиотеки для поддержки одноименного криптографического протокола. |
| **Libc** | - стандартная библиотека языка С, а именно ее BSD реализация, настроенная для работы на устройствах на базе Linux. |

Среда исполнения включает в себя библиотеки ядра, обеспечивающие большую часть низкоуровневой функциональности, доступной библиотекам ядра языка *Java*, и виртуальную машину Dalvik, позволяющую запускать приложения. Каждое *приложение* запускается в своем экземпляре виртуальной машины, тем самым обеспечивается изоляция работающих приложений от ОС и друг от друга. Для исполнения на виртуальной машине Dalvik *Java*-классы компилируются в исполняемые файлы с расширением .dex с помощью инструмента dx, входящего в состав *Android* *SDK*. DEX (Dalvik EXecutable) - формат исполняемых файлов для виртуальной машины Dalvik, оптимизированный для использования минимального объема памяти. При использовании *IDE* Eclipse и плагина *ADT* (*Android* *Development* *Tools*) *компиляция* классов*Java* в формат .dex происходит автоматически.

*Архитектура* *Android* Runtime такова, что работа программ осуществляется строго в рамках окружения виртуальной машины, что позволяет защитить *ядро* ОС от возможного вреда со стороны других ее составляющих. Поэтому код с ошибками или *вредоносное ПО* не смогут испортить*Android* и устройство на его базе, когда сработают.

На еще более высоком уровне располагается каркас приложений (*Application* Framework), *архитектура* которого позволяет любому приложению использовать уже реализованные возможности других приложений, к которым разрешен *доступ*. В состав каркаса входят следующие компоненты:

* богатый и расширяемый набор представлений (**Views**), который может быть использован для создания визуальных компонентов приложений, например, списков, текстовых полей, таблиц, кнопок или даже встроенного web-браузера;
* контент-провайдеры (**Content Providers**), управляющие данными, которые одни приложения открывают для других, чтобы те могли их использовать для своей работы;
* менеджер ресурсов (**Resource Manager**), обеспечивающий доступ к ресурсам без функциональности (не несущим кода), например, к строковым данным, графике, файлам и другим;
* менеджер оповещений (**Notification Manager**), позволяющий приложениям отображать собственные уведомления для пользователя в строке состояния;
* менеджер действий (**Activity Manager**), управляющий жизненными циклами приложений, сохраняющий историю работы с действиями, предоставляющий систему навигации по действиям;
* менеджер местоположения (**Location Manager**), позволяющий приложениям периодически получать обновленные данные о текущем географическом положении устройства.

*Application* Framework предоставляет в распоряжение приложений в ОС *Android* вспомогательный функционал, благодаря чему реализуется принцип многократного использования компонентов приложений и ОС. Естественно, в рамках политики безопасности.

И, наконец, самый высокий, самый близкий к пользователю уровень приложений. Именно на этом уровне *пользователь* взаимодействует со своим устройством, управляемым ОС *Android*. Здесь представлен набор базовых приложений, который предустановлен на ОС *Android*. Например, *браузер*, *почтовый клиент*, *программа* для отправки *SMS*, карты, календарь, *менеджер* контактов и др. *Список* интегрированных приложений может меняться в зависимости от модели устройства и версии *Android*. К этому уровню также относятся все пользовательские приложения.

Разработчик обычно взаимодействует с двумя верхними уровнями архитектуры *Android* для создания новых приложений. Библиотеки, система исполнения и *ядро* Linux скрыты за каркасом приложений.

Повторное использование компонентов других приложений приводит к идее задач в *Android*. *Приложение* может использовать компоненты другого *Android* приложения для решения задачи, например, если разрабатываемое *приложение* предполагает использование фотографий, оно может вызвать *приложение*, управляющее фотографиями и зарегистрированное в системе *Android*, выбрать с его помощью фотографию и работать с ней.

Для пополнения коллекции приложений своего мобильного устройства *пользователь* может воспользоваться приложением Google *Play*, которое позволяет покупать и устанавливать приложения с сервиса Google *Play*. Разработчики, в свою *очередь*, могут выкладывать свои приложения в этот сервис, Google *Play* отслеживает появление обновлений приложения, сообщает пользователям этого приложения об обновлении и предлагает установить его. Также Google *Play* предоставляет разработчикам *доступ* к услугам и библиотекам, например, *доступ* к использованию и отображению Google Maps.

Для установки приложения на устройствах с ОС *Android* создается *файл* с расширением \*.apk (*Android* *package*), который содержит исполняемые файлы, а также вспомогательные компоненты, например, файлы с данными и *файлы ресурсов*. После установки на устройство каждое*приложение* "живет" в своем собственном изолированном экземпляре виртуальной машины Dalvik.

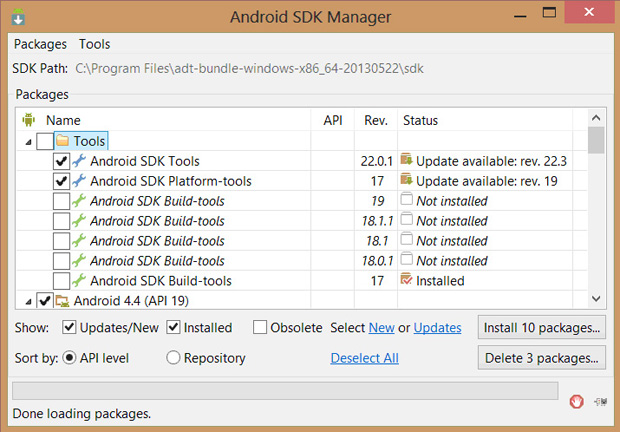
**1.3 Обзор сред программирования**

Прежде чем начать разрабатывать приложения под *Android*, рассмотрим существующие инструменты, подходящие для этих целей. Можно выделить необходимые инструменты, без которых разработка мобильных приложений под *Android* просто невозможна. С другой стороны, существует большое количество вспомогательных систем, в какой-то мере упрощающих процесс разработки.

К обязательным инструментам относится *Android* *SDK* - набор средств программирования, который содержит инструменты, необходимые для создания, компиляции и сборки мобильного приложения.

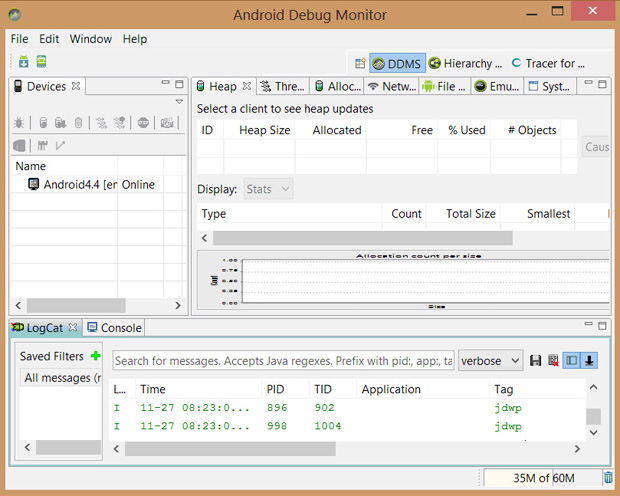
Рассмотрим кратко наиболее важные инструменты, входящие в состав *Android* *SDK*:

* **SDK Manager** - инструмент, позволяющий загрузить компоненты Android SDK. Показывает пакеты Android SDK и их статус: установлен (Installed), не установлен (Not Installed), доступны обновления (Update available).

[](http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_02.jpg)

**Рис. 1.2.**Android SDK Manager

* **Debug Monitor** - самостоятельный инструмент, предоставляющий графический интерфейс к нескольким инструментам, предназначенным для анализа и отладки Android приложений:
  + DDMS (Dalvik Debug Monitor Server) предоставляет услуги переброса портов, захват экрана устройства, информацию о потоках и динамической памяти устройства, вывод информации о действиях Android в реальном времени (logcat) и многое другое.
  + Hierarchy Viewer позволяет отлаживать и оптимизировать пользовательский интерфейс Android приложения.
  + Tracer for OpenGL ES - инструмент для анализа OpenGL|ES кода, используемого в мобильном приложении, позволяет захватывать команды OpenGL|ES и демонстрировать их по отдельным кадрам, что помогает понять как исполняются графические команды.

[](http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_03.jpg)

**Рис. 1.3.**Окно инструмента Monitor

* **Android Emulator (emulator)** - виртуальное мобильное устройство, которое создается и работает на компьютере разработчика, используется для разработки и тестирования мобильных приложений без привлечения реальных устройств.
* **AVD Manager** - предоставляет графический интерфейс для создания виртуальных Android устройств (AVDs), предусмотренных Android Emulator, и управления ими.

(В ЛР№1 подробно рассматривается создание и использование виртуального устройства).

* **Android Debug Bridge (adb)** - гибкий инструмент, позволяющий управлять состоянием эмулятора или реального Android устройства, подключенного к компьютеру. Также может использоваться для установки Android приложения (.apk файл) на реальное устройство.

Мы рассмотрели основные инструменты, входящие в состав *Android* *SDK*, разумеется, не все и недостаточно подробно. Для более серьезного изучения инструментов имеет смысл обратиться к сайту разработчиков (<http://developer.android.com/tools/help/index.html>). Для разработки мобильных приложений под *Android* уверенного владения инструментами из *SDK* вполне достаточно. Если же возникают какие-то вопросы, дополнительные инструкции *по* созданию проектов, компиляции, запуску из командной строки содержатся в руководстве от Google (<http://developer.android.com/training/basics/firstapp/index.html>).

В современных условиях разработка *ПО* в большинстве случаев ведется с использованием интегрированных сред разработки (*IDE*). *IDE* имеют несомненные достоинства: процесс компиляции, сборки и запуска приложения обычно автоматизирован, в связи с чем для начинающего разработчика создать свое первое *приложение* труда не составляет. Но чтобы заниматься разработкой всерьез, необходимо потратить силы и время на изучение возможностей самой среды. Рассмотрим *IDE*, пригодные для разработки под Android1.

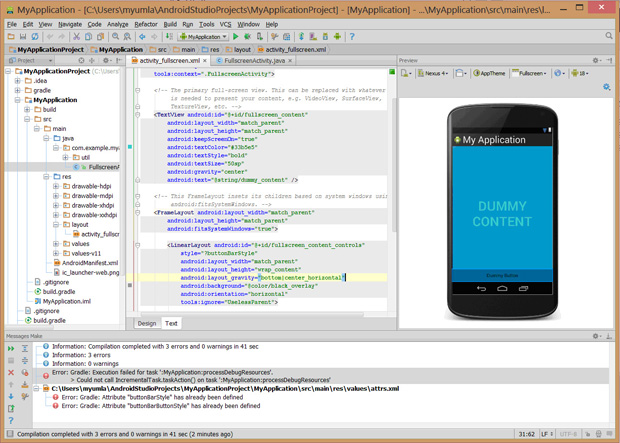
Для начала поговорим о двух средах разработки, которые рекомендует Google: *Android* *IDE* (*ADT*) и *Android* Studio.

**Android IDE** - *среда разработки* под *Android*, основанная на Eclipse. Предоставляет интегрированные инструменты для разработки, сборки и отладки мобильных приложений. В данном курсе *Android* *IDE* выбрана в качестве основной среды разработки. Возможности этой среды более подробно рассмотрены в первой лабораторной работе. Также там даны рекомендации *по* установке и настройке среды, созданию и запуску первого приложения как на эмуляторе, так и на реальном устройстве.

**Android Studio** - *среда разработки* под *Android*, основанная на IntelliJ *IDEA*. Подобно *Android* *IDE*, она предоставляет интегрированные инструменты для разработки и отладки. Дополнительно ко всем возможностям, ожидаемым от IntelliJ, в *Android* Studio реализованы:

* поддержка сборки приложения, основанной на Gradle;
* специфичный для Android рефакторинг и быстрое исправление дефектов;
* lint инструменты для поиска проблем с производительностью, с юзабилити, с совместимостью версий и других;
* возможности ProGuard (утилита для сокращения, оптимизации и обфускации кода) и подписи приложений;
* основанные на шаблонах мастера для создания общих Android конструкций и компонентов;
* WYSIWYG редактор, работающий на многих размерах экранов и разрешений, окно предварительного просмотра, показывающее запущенное приложение сразу на нескольких устройствах и в реальном времени;
* встроенная поддержка облачной платформы Google.

Загрузить последнюю версию *Android* Studio, а также получить рекомендации *по* установке, настройке и началу работы можно тут:<http://developer.android.com/sdk/installing/studio.html>.

[](http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_04.jpg)

**Рис. 1.4.**Среда разработки Android Studio

Перейдем к рассмотрению других инструментов, пригодных для разработки мобильных приложений под *Android*. Начнем с инструментов от Intel - Intel XDK и Intel Beacon Mountain.

**Intel XDK** позволяет легко разрабатывать кроссплатформенные мобильные приложения; включает в себя инструменты для создания, отладки и сборки *ПО*, а также эмулятор устройств; поддерживает разработку для *Android*, Apple iOS, Microsoft *Windows* 8, Tizen; поддерживает языки разработки: HTML5 и JavaScript.

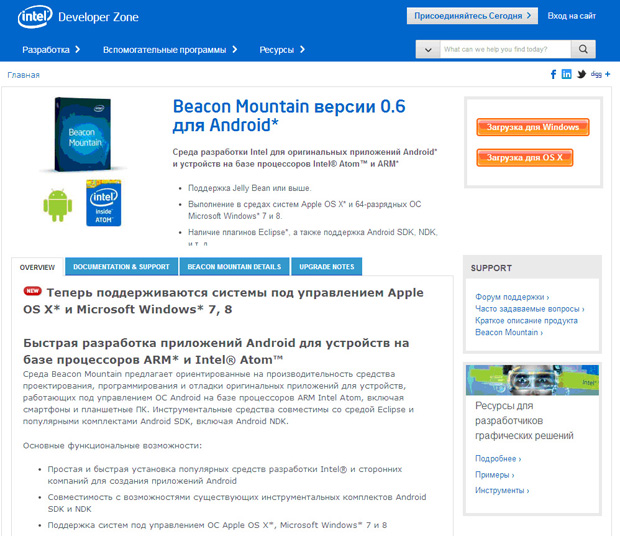
Последняя тема данного курса полностью посвящена изучению нового поколения инструментальных средств разработки мобильных HTML5-приложений и Intel XDK, предполагается разработка мобильного приложения с использованием этих инструментов.

**Intel Beacon Mountain** - *среда разработки*, позволяющая создавать приложения для устройств, работающих под управлением ОС *Android*. Предоставляет инструменты необходимые для проектирования, разработки, отладки и оптимизации приложений под *Android*. Освобождает разработчика от необходимости поддерживать систему разработки в актуальном состоянии, следит за обновлениями и добавляет их в среду разработки *по* мере появления. Поддерживает разработку для целевых платформ на основе процессоров Intel *Atom* и *ARM*.

Beacon Mountain построена на основе *Android* *IDE* (Eclipse, *Android* *ADT*, *Android* *SDK*), для более серьезной разработки и оптимизации добавлены следующие инструменты Intel:

* **Intel\* Hardware Accelerated Execution Manager (Intel\* HAXM)** - аппаратно поддерживаемый процессор виртуализации, использующий технологию виртуализации Intel\* (Intel\* VT) для ускорения работы эмулятора в среде разработки.
* **Intel\* Graphics Performance Analyzers (Intel\* GPA) System Analyzer** поддерживает мобильные устройства с процессором Intel Atom под управлением ОС Android. Позволяет разработчикам оптимизировать загруженность системы при использовании процедур OpenGL, предоставляя возможность получать множество системных метрик в реальном времени, отображающих загруженность CPU, GPU и OpenGL ES API. Разработчик может запустить несколько графических экспериментов для выявления узких мест в обработке графики.
* **Intel\* Integrated Performance Primitives (Intel\* IPP) Preview** - библиотека оптимизированной обработки данных и изображений, поддерживающая мобильные устройства с платформой Intel под управлением ОС Android. Preview версия является частью полной версии Intel IPP, которая тоже поддерживает ОС Android.
* **Intel\* Threading Building Blocks (Intel\* TBB)** - широко используемая, признанная библиотека шаблонов С++ для создания масштабируемых приложений и увеличения производительности. Поддерживает мобильные устройства с платформой Intel под управлением Android. Проверенные алгоритмы позволяют разработчикам эффективно распараллелить С++ мобильные приложения, что повышает производительность при снижении энергетических затрат.

Загрузить Intel Beacon Mountain можно *по* ссылке <http://software.intel.com/ru-ru/vcsource/tools/beaconmountain>

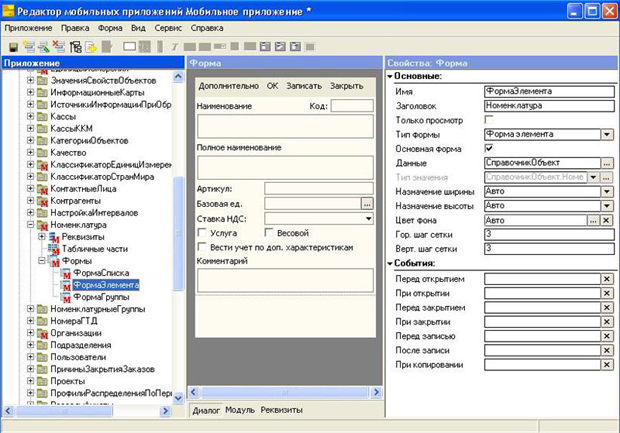
[](http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_05.jpg)

**Рис. 1.5.**Страница поддержки Intel\* Beacon Mountain

Нельзя обойти вниманием *инструментарий* Marmalade *SDK*.

**Marmalade SDK** - кроссплатформенное *SDK* от Ideaworks3D Limited. Представляет собой набор библиотек, образцов, инструментов и документации, необходимых для разработки, тестирования и развертывания приложений для мобильных устройств. Используется, в основном, для разработки игр. Многие получившие признание игры, такие как *Cut* the Rope и Plants vs. Zombies, были разработаны с использованием этого программного средства. К сожалению, Marmalade *SDK* представляет собой проприетарное *программное обеспечение* (самая дешевая*лицензия* $15 в месяц) и не может быть рекомендована в данном учебном курсе, но читатель может самостоятельно попробовать бесплатную 30-дневную версию, доступную *по* ссылке <https://www.madewithmarmalade.com/free-trial>.

Нельзя не сказать об отечественных разработках. Например, компания 1С идет в ногу со временем, версия платформы 1С 8.3 позволяет разрабатывать мобильные приложения. *Программный продукт* **"1С:Предприятие 8. Расширение для карманных компьютеров"**обеспечивает возможность работы с данными информационных баз 1С:Предприятия 8 на мобильных устройствах (карманных компьютерах, коммуникаторах, терминалах сбора данных), а также на персональных компьютерах (в том числе ноутбуках), не имеющих прямого доступа к информационным базам 1С:Предприятия 8.

[](http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_06.jpg)

**Рис. 1.6.**Редактор мобильных приложений 1С

Разумеется, данный *программный продукт* имеет очень узкую сферу применения, однако в некоторых случаях может являться наиболее удачным решением. Подробности *по* ссылке <http://v8.1c.ru/overview/Term_000000401.htm#1>

### 1.4 Эмуляторы

#### Эмуляция. Стандартный эмулятор Android

**Эмуляция** (англ. *emulation*) в вычислительной технике - комплекс программных, аппаратных средств или их сочетание, предназначенное для копирования (или *эмулирования*) функций одной вычислительной системы (*гостя*) на другой, отличной от первой, вычислительной системе (*хосте*) таким образом, чтобы эмулированное поведение как можно ближе соответствовало поведению оригинальной системы (*гостя*). Целью является максимально точное воспроизведение поведения в отличие от разных форм компьютерного моделирования, в которых имитируется поведение некоторой абстрактной модели ([Википедия](http://ru.wikipedia.org/wiki/%D0%AD%D0%BC%D1%83%D0%BB%D1%8F%D1%86%D0%B8%D1%8F)).

Эмулятор - виртуальное мобильное устройство, которое запускается на компьютере. При помощи эмулятора можно разрабатывать и тестировать приложения без использования реальных устройств. На рисунке 1.7 приведен пример запущенного стандартного эмулятора. Подробно работа с эмуляторами рассмотрена в лабораторной работе.

|  |  |
| --- | --- |
| [http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_07_1sm.jpg](http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_07_1.jpg) | [http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_07_2sm.jpg](http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_07_2.jpg) |
| **Рис. 1.7.** Эмулятор Android SDK в процессе запуска и приложение "Hello, world!" | |

К достоинствам использования эмуляторов можно отнести простоту их использования и нулевую стоимость. Разработчику не нужно покупать огромное количество устройств с различными характеристиками, чтобы проверить работоспособность приложения на различных смартфонах. Достаточно создать несколько эмуляторов с требуемыми характеристиками и запустить на них приложение.

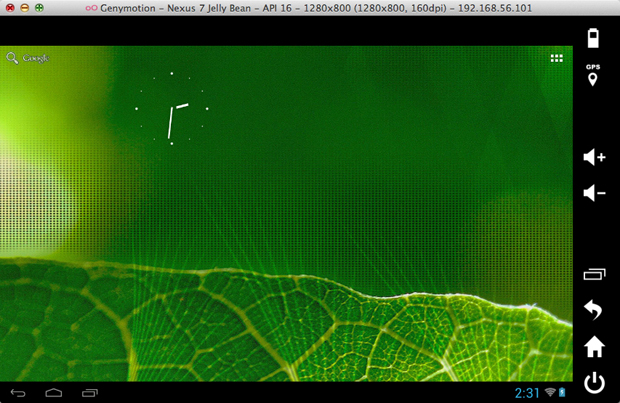
К сожалению, эмуляторы имеют и ряд недостатков:

* Требуют много системных ресурсов.
* Из-за различий в архитектуре процессоров компьютера и смартфона медленно запускаются. Современные персональные компьютеры построены на архитектурах x86 и x64, а большинство процессоров смартфонов на Android - ARM. Процесс эмуляции одной архитектуры на другой чрезвычайно сложен и происходит довольно медленно.
* В некоторых случаях стандартного эмулятора недостаточно. Речь идет о возможностях смартфонов, которыми обычные компьютеры не обладают (например, наличие датчика gps или акселерометра). В таких случаях полноценную отладку можно провести только с использованием реального устройства.

#### Альтернативные эмуляторы

Стандартный эмулятор, поставляемый вместе с Android SDK, не устраивает многих. Существуют проекты, поддерживающие разработку и развитие альтернативных эмуляторов. В качестве примера можно привести Genymotion (см. [рис. 1.8](http://www.intuit.ru/studies/courses/12643/1191/lecture/21980?page=4#image.1.8)) - быстрый эмулятор Android (по мнению его разработчиков). Он содержит предварительно настроенные образы Android (x86 с аппаратным ускорением OpenGL). Genymotion доступен для Linux, Windows и Mac OS X и требует для своей работы VirtualBox. Иными словами, Genymotion представляет собой виртуальную машину с установленной ОС Android, которую пользователь запускает так же, как и другие виртуальные машины. Проблема высокого потребления системных ресурсов, конечно, не исчезает, однако скорость запуска существенно увеличивается.

В настоящее время активно развивается.

[](http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/1/files/01_08.jpg)

**Рис. 1.8.**Альтернативный эмулятор Genymotion

### 1.5 Возможности отладки на реальных устройствах

Разработанное *приложение* можно запустить на реальном устройстве, например, на смартфоне. Для этого необходимо проделать предварительную работу.

Для запуска приложений, разработанных в *Android* *IDE*, необходимо:

* Настроить устройство (включить режим отладки по USB).
* Настроить компьютер (для Windows необходимо установить нужный драйвер вручную, нужны права администратора).
* Настроить среду и запустить проект на устройстве.

Подробности отладки на реальных устройствах описаны в лабораторной работе.

### 1.6 Примеры приложений

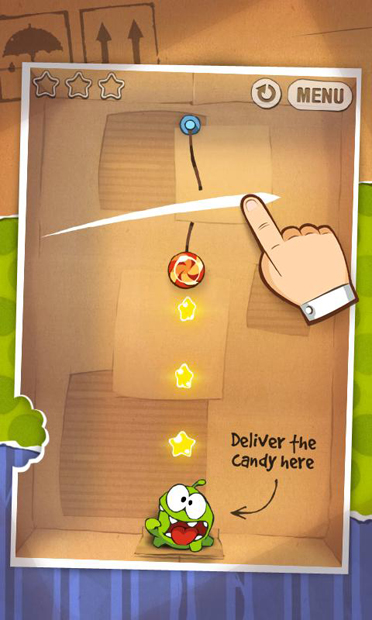
Google *Play* - это магазин приложений от Google, позволяющий владельцам устройств с операционной системой *Android* устанавливать и приобретать различные приложения. Учётная *запись* разработчика, которая даёт возможность публиковать приложения, стоит $25. В настоящее время Google *Play* насчитывает более миллиона различных приложений, каждый месяц пользователями загружается несколько миллиардов. Разумеется, далеко не все из них высокого качества и поддерживаются разработчиками, встречается и вредоносное *программное обеспечение*.

В настоящий момент доступно более 30 различных категорий приложений. Внутри каждой категории приложения упорядочены на основании рейтинга, отзывов, количества скачиваний, страны распространения и других факторов.

Время от времени редакция Google *Play* собирает коллекции приложений или игр, основанных на теме или сезонном событии. Коллекции пользуются популярностью у клиентов за счет своевременности и актуальности.

Приведем примеры интересных и удобных приложений, заслуженно удерживающих высокие места в своих категориях уже долгое время.

Популярное игровое *приложение* *Cut* the Rope позволяет разобраться в правилах игры прямо в ее процессе и не требует чтения сложных инструкций (см. [рисунок 1.9](http://www.intuit.ru/studies/courses/12643/1191/lecture/21980?page=4#image.1.9)). Идея игры предельно проста - в коробке сидит маленький зелёный монстр Ам Ням, которого надо кормить леденцами. Леденцы болтаются на веревках, и их надо правильно перерезать, чтобы леденец попал точно в рот Ам Няма. *По* ходу игры сложность уровней возрастает, появляются дополнительные препятствия. Попутно надо собирать звездочки, которые позволяют открывать новые локации.



**Рис. 1.9.**Первый уровень игры Cut the Rope

Если вам нужен интерактивный помощник, который способен понимать сделанные устно указания и напоминать о делах в нужное время,*приложение* "Помнить все" (см. [рис. 1.10](http://www.intuit.ru/studies/courses/12643/1191/lecture/21980?page=4#image.1.10)) - то, что вам нужно. Используя библиотеку для распознавания речи, оно анализирует полученную информацию, отображает ее в виде текста, который при необходимости можно исправить, и устанавливает время для напоминания.



**Рис. 1.10.**Приложение "Помнить все" распознает русскую речь

### Контрольные вопросы:

1. Каково устройство платформы Android?
2. Что представляет собой Android SDK?
3. Назовите основные средства разработки под Android.
4. Перечислите достоинства и недостатки эмуляторов Android.
5. Выясните объем продаж мобильных устройств с ОС Android.
6. Какая версия платформы наиболее популярна в настоящее время?