Android Programing

Gavrilut Dragos

Administrative

Final grade for the Android course is computed using Gauss over the total points accumulated.

- Maximum 100 points from the laboratory examination
- Maximum 60 points at the final examination (course)

The minimum number of points that one needs to pass this exam:

- Minimum 50 points accumulated from the laboratory examination
- Minimum 20 points from the final examination (course)

- Course page: https://sites.google.com/site/fiiandroidprogramming/home

History

- 2003 October – Android Inc, is founded
- 2005 August – Google Inc. buys Android Inc for 50.000.000 USD
- 2007 – Android becomes an open-source project under Apache license
- 2008 October – first device with Android OS is release
- 2010 January → Nexus One (develop by HTC)
- 2010 September → Nexus S (develop by Samsung)
- 2011 November → Nexus Galaxy (develop by Samsung)
- 2012 November → Nexus 4 (develop by LG)
- 2013 October → Nexus 5 (develop by LG)
- 2014 October → Nexus 6 (develop by Motorola)
- 2016 – Pixel phone is announced (currently on Pixel 5 release on Sep.2020)
- 2017 - Google Glass Enterprise Edition
- 2017 – Google Announces first support for Kotlin language
- 2017 – Android Wear (first release) for smart watches
- 2019 - Google Glass Enterprise Edition 2
- 2020 – Android Things reaches EOL

Android versions

Release date	Version	Code Name	API Level
2008.Sep	1.0	-	1
2009.Feb	1.3	Petite Four	2
2009.Apr	1.5	Cupcake	3
2009.Sep	1.6	Donut	4
2009.Oct	2.0	Eclair	5
2009.Dec	2.0.1	Eclair	6
2010.Ian	2.1	Eclair	7
2010.May	2.2	Froyo	8
2010.Dec	2.3 – 2.3.2	Gingerbread	9
2011.Feb	2.3.3 – 2.3.7	Gingerbread	10
2011.Feb	3.0	Honeycomb	11
2011.May	3.1	Honeycomb	12

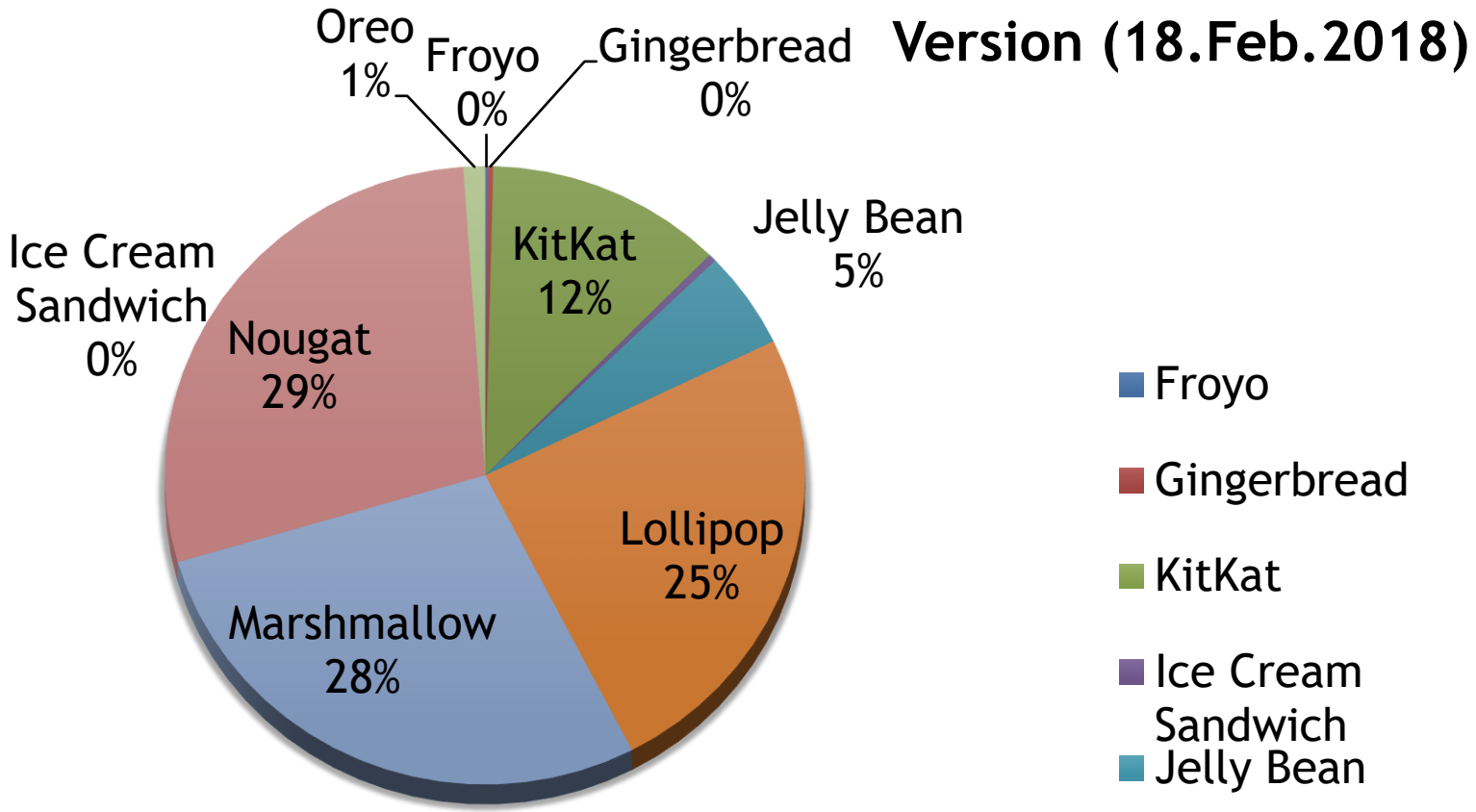
Android versions

Release date	Version	Code Name	API Level
2011.July	3.2	Honeycomb	13
2011.Oct	4.0-4.0.2	Ice Cream Sandwich	14
2011.Dec	4.0.3 – 4.0.4	Ice Cream Sandwich	15
2012.Jul	4.1	Jelly Bean	16
2012.Nov	4.2	Jelly Bean	17
2013.Jul	4.3	Jelly Bean	18
2013.Sep	4.4	Kit Kat	19
2014.July	4.4W	Kit Kat	20
2014.Nov	5.0	Lollipop	21
2015.Mar	5.1	Lollipop	22
2015.Oct	6.0	Marshmallow	23
2016.Aug	7.0 – 7.1	Nougat	24/25
2017.Aug	8.0 – 8.1	Oreo	26/27

Android versions

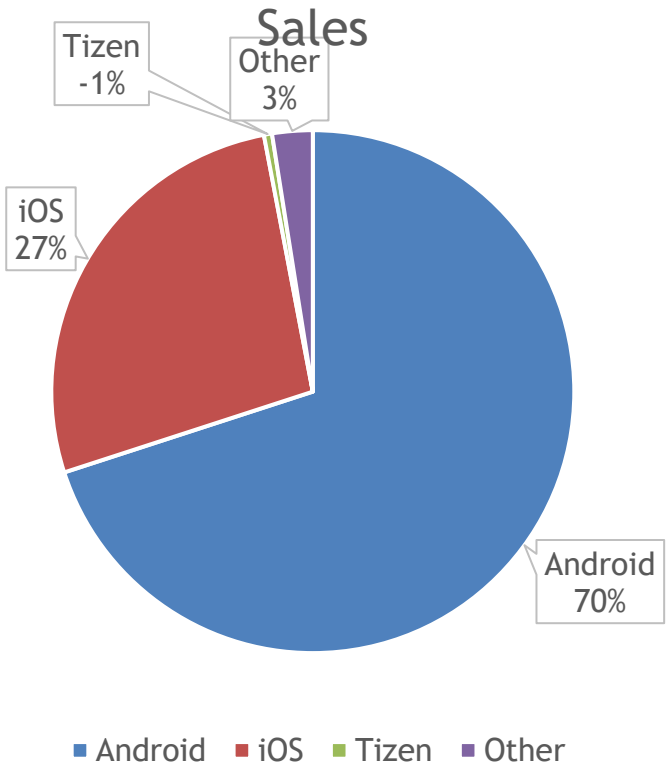
Release date	Version	Code Name	API Level
2018.August	9	Pie	28
2019.September	10	Android 10	29
2020.September	11	Android 11	30

Android Distribution

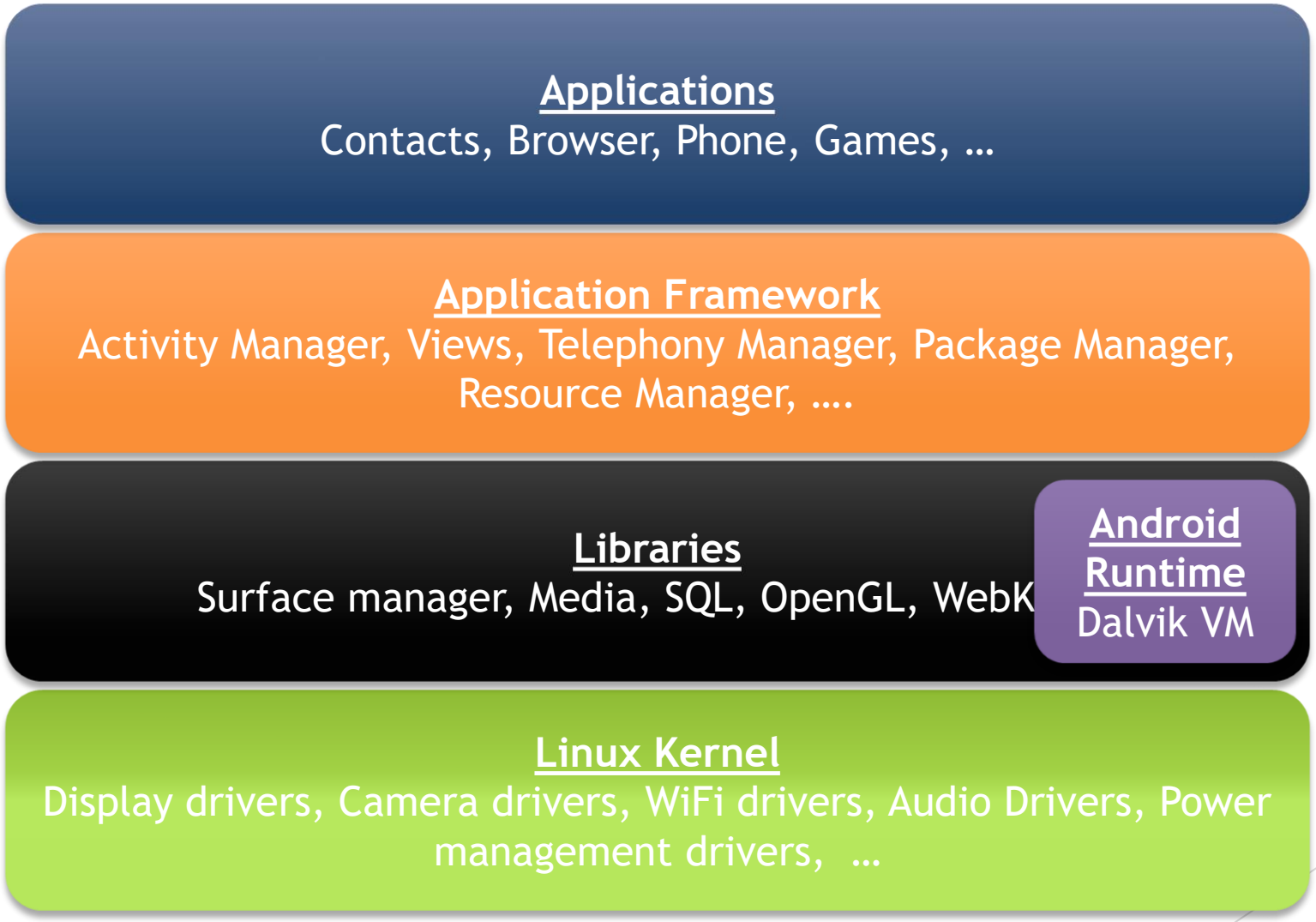


Google has stopped providing this information on their dashboard.

Market share



Android architecture



Android architecture

- Hardware platform
  - ARM
  - X86 (Google TV, …)
  - i.MX
  - Intel

Mobile architecture

- Touch Screens
- SMS & Phone
- GPS
- Flash Drives
- NFC
- WiFi
- DLNA
- ….

Legal issues

- Google vs Oracle
  - Java structure
- Apple vs Samsung
  - Design
- Apple & Microsoft vs HTC & Samsung
- Google buys Motorola

File Hierarchy

- Different file systems (YAFFS, EXTx, proprietary (Samsung RFS), F2FS, F2FS, JFFS2, …)
- Partitions:
  - /cache
  - /system
  - /sdcard
  - /mnt
  - /sys
  - /data
  - /root
  - /dev
  - …

File Hierarchy

- /cache
  - Cached files and data
- /sdcard
  - Contains application data, pictures, etc
  - Some applications use this partition as a way to record different data regarding their instalation (install date, etc)
- /system
  - Includes Android OS files (except for kernel) such as libraries, fonts, default applictions (email, browser, phone, …), system sounds, linux executables for different commands (ls, rm, su, …), …

File Hierarchy

- /data
  - User specific data (contacts, messages, settings, …)
  - Private data and libraries for every application installed (by package)
  - Installed applications

APK format

- APK = **A**pplication **P**ackage **F**ile
- ZIP archive
  - **\META-INF**
    - MANIFEST.MF
    - CERT.RSA
    - CERT.SF (SHA-1 digest for MANIFEST.MF)
  - **\lib**
    - \armeabi
    - \armeabi-v7a
    - \x86
    - \mips
  - **\res**
    - drawable-{xxx}
    - raw-{xxx}
    - layout -{xxx}
    - menu
  - **\assets**
  - classes.dex
  - resources.arsc
  - AndroidManifest.xml

Dalvik

- Register-based VM
- Sandbox
- A **DEX** file contains all the information required for the Dalvik VM to execute the code (libraries, endianess, …)
- Can run native code.
- “Every Android application runs in its own process, with its own instance of the Dalvik virtual machine.”

DEX files

- Header (CheckSum, SHA, …)
- String Indexes
- Type Indexes
- Prototype Indexes
- Field Indexes
- Method Indexes
- Class Definitions
- Data (dex code , strings, classes, …)

Zygote process

- Use to increase the start time of a dalvik VM process
- Shares constant data (libraries) between instances of VM processes

```
def ZygoteStart:
    while (true)
        if (new app is requested)
            fork()
        endif
    endwhile
enddef
```

- Uses “copy-on-write” to copy modified memory to a spawn child

Aplications

- Each Android Application runs as a linux process
- Each Android Application has multiple components:
  - Activities
  - Services
  - Content Providers
  - Broadcast Receivers
- Each Android Application can start another Android Application components (use an activity from email application to send email). This can be done using **Intent** object.

Aplications

- Each Android Application has its own process. Each process has a rank (importance) in Android. The more important a process is, the less is the chance that it will be killed by the system.
- There are 5 ranks for processes:
  1. Forenground process
     - Has an Activity that users interacts with
     - Has a Service that interacts with a Forenground process
     - Has a Service that runs in forenground
     - Has an active Broadcast Receiver
  2. Visible process
     - Has an Activity that is in background (paused)
     - Has a Service that is linked to a background activity
  3. Service process
     - Has a Service
  4. Background process
     - Has an Activity with a process that was stopped
  5. Empty process
     - Does not have any components. It is maintained for caching purposes.

Aplications

- To execute an Android Application the system checks **AndroidManifest.xml** file
- AndroidManifest file contains:
  - Permissions
  - List of activities
  - List of services
  - List of receivers
  - List of providers

```
<?xml version="1.0" encoding="utf-8"?>
<manifest ... >
    <application android:icon="@drawable/app_icon.png“>
        <activity android:name=“com.myapp.myActivity"
                  android:label="@string/HelloWorld">
        </activity>
        ...
    </application>
</manifest>
```

Aplications

```
<application android:allowTaskReparenting=["true" | "false"]
        android:description="string resource"
        android:hasCode=["true" | "false"]
        android:hardwareAccelerated=["true" | "false"]
        android:icon="drawable resource"
        android:label="string resource"
        android:name="string"
        android:permission="string"
        android:process="string"
        android:theme="resource or theme"
. . .
</application>
```

Activity Life Cycle