Why are Mobile Computing Systems Different from Desktop Systems?

- Heterogeneous hardware
 - Mobiles have far more variety than desktop systems, with >10 companies selling smartphones all over the world
 - o However, all smartphones today use one System-on-chip among 4 companies:
 - Qualcomm (Snapdragon)
 - MediaTek (HeliOS)
 - Samsung (Exynos)
 - Huawei (Kirin)
 - Apple
 - o All the smartphones use ARM instruction-set architecture
 - Uses RISC instructions, so sizes of programs are larger, but processors are more power-efficient
 - CPUs have big-little core architecture, i.e. some cores are more powerful in compute frequency but also more power-hungry
 - Two different frequencies usually reported in the specs
 - Less urgent tasks should be given to the less powerful cores
 - In addition, across smartphones there is a huge variety in the type of hardware available
 - o Additional Reference: Whitepaper from Samsung on Big-Little Architecture
- Power constraint
 - All smartphones run on battery
 - Capacity of battery has not improved
 - O What consumes battery?
 - Sensing gyroscope, GPS, touchscreen, etc
 - Computation CPU, GPU, etc
 - Display screens increasing
 - Network cellular, WiFi (high power consumption; very important topic; but covered in wireless networks course)
 - How do mobile systems manage power?
 - By reducing the frequency at which the power is used; this concept is known as dynamic voltage and frequency scaling
 - By being context-aware, i.e. disabling sensing when it is not needed
 - Additional optimizations of display techniques such as reducing brightness at low-battery, etc
 - Additional Reference:
 https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8930492 (Page 11-14)
- Security constraint

- Security is more important for smartphones than desktop devices, as more private data is present
- Applies sandboxing on apps, i.e. data of one app cannot be accessed by other apps
- Additional constraint on apps to ask for permission to access sensors and storage data
- Sandboxing applied using the concept of containers given by the Linux kernel
- Reference: https://source.android.com/docs/security/app-sandbox
- Additional region of memory and separate processor present for highly sensitive data like fingerprints, etc: https://source.android.com/docs/security/features/trusty

Android: Concept of programming in layers

