

CE191

Operating Systems and Systems Programming Introduction

Intro To OS Concepts

September 2019

Team-TAs



Amir Hossein Sorouri

- Master of SE
- DSLab



Vahid Mohseni

- Bachelor of CE
- DSLab



Hamid Hadian

- Master of SE
- DSLab



Ehsan AliAkbar

- Bachelor of CE
- DSLab



<https://os-course.github.io>



[Iran University ST OS191](#)



Check out [`github.com/os-course`](https://github.com/os-course)

Grading

- ▶ Grades will be determined roughly as follows:
- ▶ Quiz 15%
- ▶ Homework 25%
- ▶ Projects 40%
- ▶ Participation & Collaboration 20%

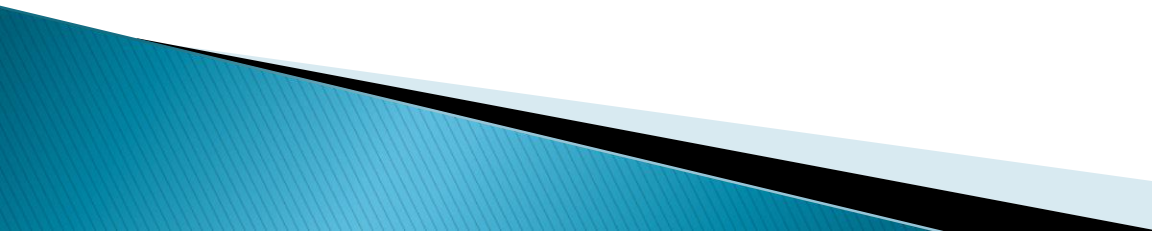
Projects and Homework

- ✓ Document
- ✓ Code

Prerequisites

- ▶ Familiarity with Linux
- ▶ Knowledge of C, and data structures
- ▶ TAs will spend some time reminding some of the prerequisites
- ▶ Bring your own laptop

Sources

- ▶ **Modern Operating Systems by Tanenbaum A.**
 - ▶ Operating Systems: three easy pieces (OSTEP) by Remzi H.
 - ▶ XV6 for projects and homework.
 - ▶ Write Your Own Operating System (WYOOS)
- 

Environment

- ▶ [Linux – Ubuntu 16.04](#)
- ▶ ubuntu-16.04.6-server-amd64.iso , ubuntu-16.04.6-desktop-i386.iso
- ▶ [Vmware](#)
- ▶ Install gcc for compile C programs
- ▶ How to Install Ubuntu 16.04 on Vmware : [tutorial](#) , [video](#)

Collaboration Policy

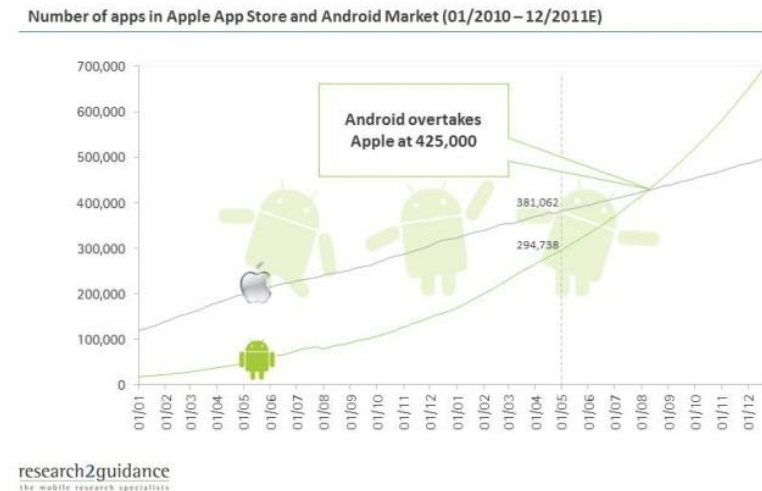
- ▶ Explaining a concept to someone in another group
 - ▶ Discussing algorithms/testing strategies with other groups
 - ▶ Helping debug someone else's code (in another group)
 - ▶ Bring your own laptop
-
- ▶ Sharing code or test cases with another group
 - ▶ Copying OR reading another group's code or test cases
 - ▶ Copying OR reading online code or test cases
 - ▶ Copying OR reading project of senior students
 - ▶ Cheating in Exams and Quizzes

Operating Systems at the heart of it all ...

- **Make the incredible advance in the underlying technology available to a rapid evolving body of applications.**
 - Processing, Communications, Storage, Interaction, Protected Sharing

- **The key building blocks**

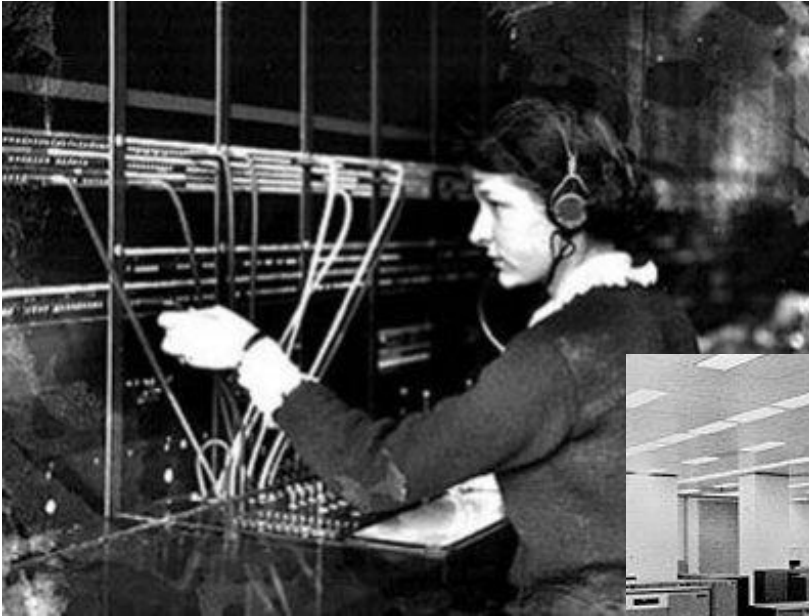
- Processes, Scheduling
- Concurrency, Coordination
- Address spaces, Translation
- Protection, Isolation, Sharing, Security
- Communication, Protocols
- Persistent storage, transactions, consistency, resilience
- Interfaces to all devices



Why we should learn OS?

- **Some of you will may design and build operating systems or components of them.**
 - Perhaps more now than ever
- **Many of you will create systems that utilize the core concepts in operating systems.**
 - Whether you build software or hardware
 - The concepts and design patterns appear at many levels
- **All of you will build applications, etc. that utilize operating systems**
 - The better you understand their design and implementation, the better use you'll make of them.

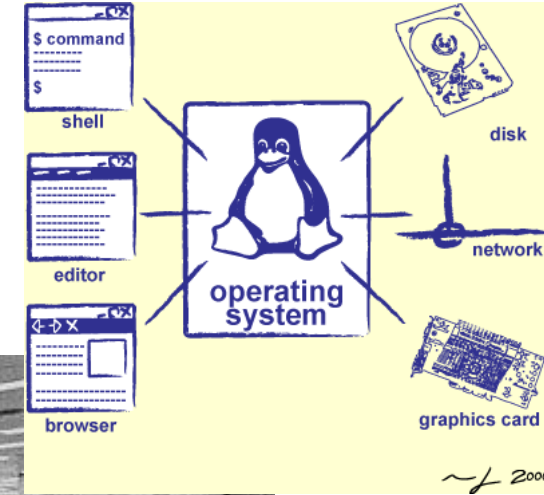
Operator ...



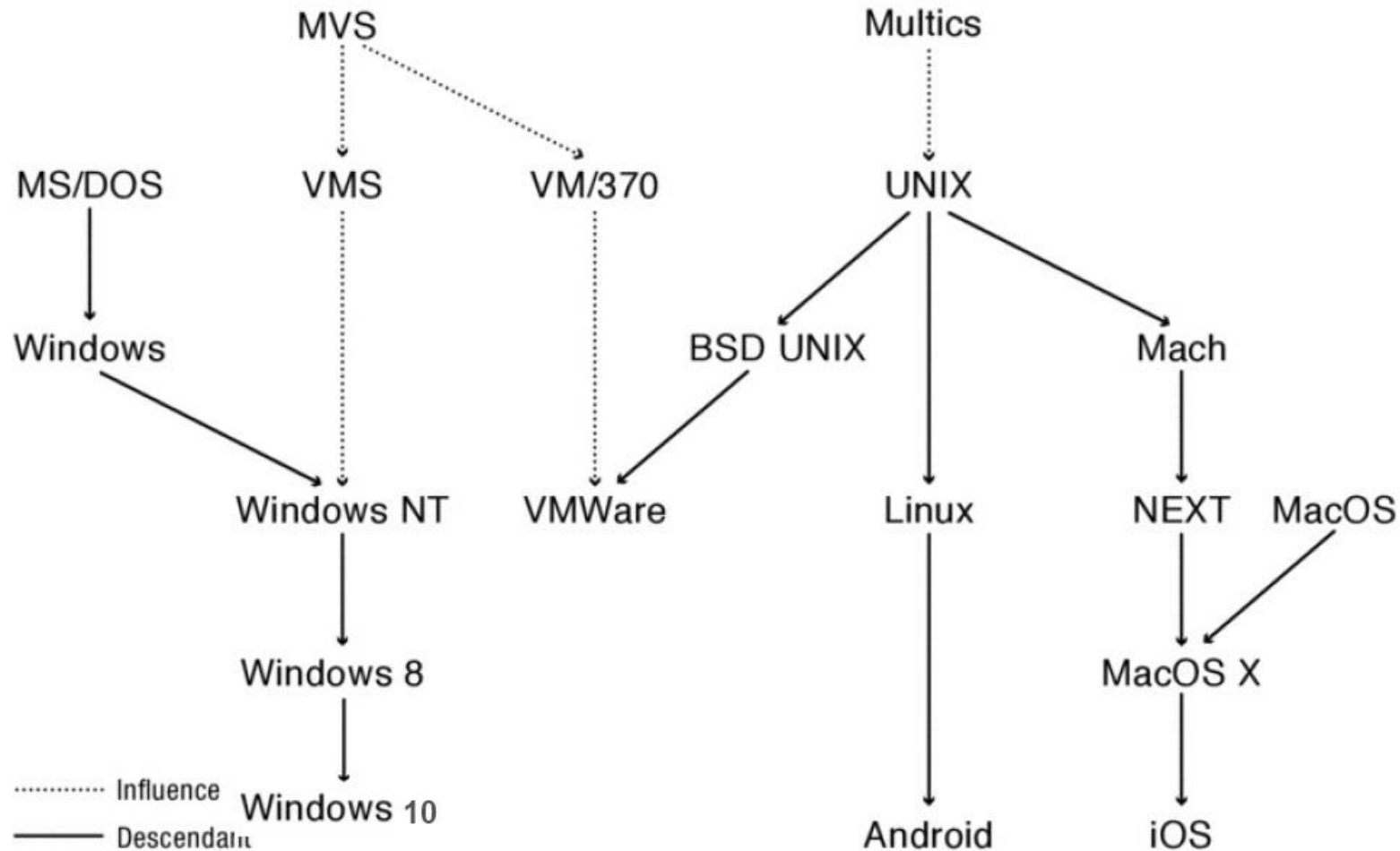
Switchboard Operator



Computer Operators



Genealogy of several modern operating systems



Time-sharing operating systems

What is an Operating System?

► Illusionist



- Provide clean, easy to use abstractions of physical resources
 - Infinite memory, dedicated machine
 - Higher level objects: files, users, messages
 - Masking limitations, virtualization