

CS 303 – Operating Systems

Chapter 0

8.1.2019

Nitin Auluck

Room 106, S. Ramanujan Block (CSE)

nitin@iitrpr.ac.in

www.iitrpr.ac.in/cse/nitin

Welcome to the course!

- Hello & welcome to CS 303 operating systems!
- Let me talk a bit about myself.
- I am Nitin Auluck, from the CSE Department here at IIT Ropar.
- I have been here since 2010 (9+ years).
- Prior to that, I was in USA from 1999 – 2010.
 - First as a PhD student at the University of Cincinnati.
 - Later as a faculty member at Quincy University.

Our Research

- Our group does research in the following topic of operating systems: **scheduling in distributed systems and fog/edge networks**.
 - Will be discussing scheduling in Chapter 5.
- Research problem:
- Given a bunch of user programs & CPU resources, coming up with algorithms for scheduling the programs on the CPUs.
- Different performance criteria to be met – throughput (# jobs/unit time), fairness, response time, deadlines, etc..
- With the reducing costs of hardware, we are seeing multicore CPU(s) in desktops, laptops, & even in phones.
- **How many CPUs in a laptop, in the iPhone?**
- **iPhone X – 6, One plus Six – 8, laptops – 4/8/16 (much better than before).**
- **More CPUs leads to better performance. How?**

Our Research

- From CSL 211, we know that each program has a set of instructions, each of which are run on the CPU.
- With multiple CPUs, we improve the program efficiency.
- At a point in time, multiple programs can run in parallel (more work done per unit time), at least theoretically.
- Resource sharing (memory, IO devices) become performance bottlenecks.
- **An example: working on a large programming project vs. working on the same project with a group.**
- **5 programmers in a group may not necessarily give 5 time improvement in throughput.**

My Research

- Our research group is looking at scheduling algorithms for large distributed systems, such as Facebook, Twitter, Google web services.
- Computing nodes/servers are distributed across the globe.
- Connected using fast networking technologies: infiniband, 4G, 5G.
- **Question: how do we schedule user requests so that an acceptable quality of service for user requests may be provided?**

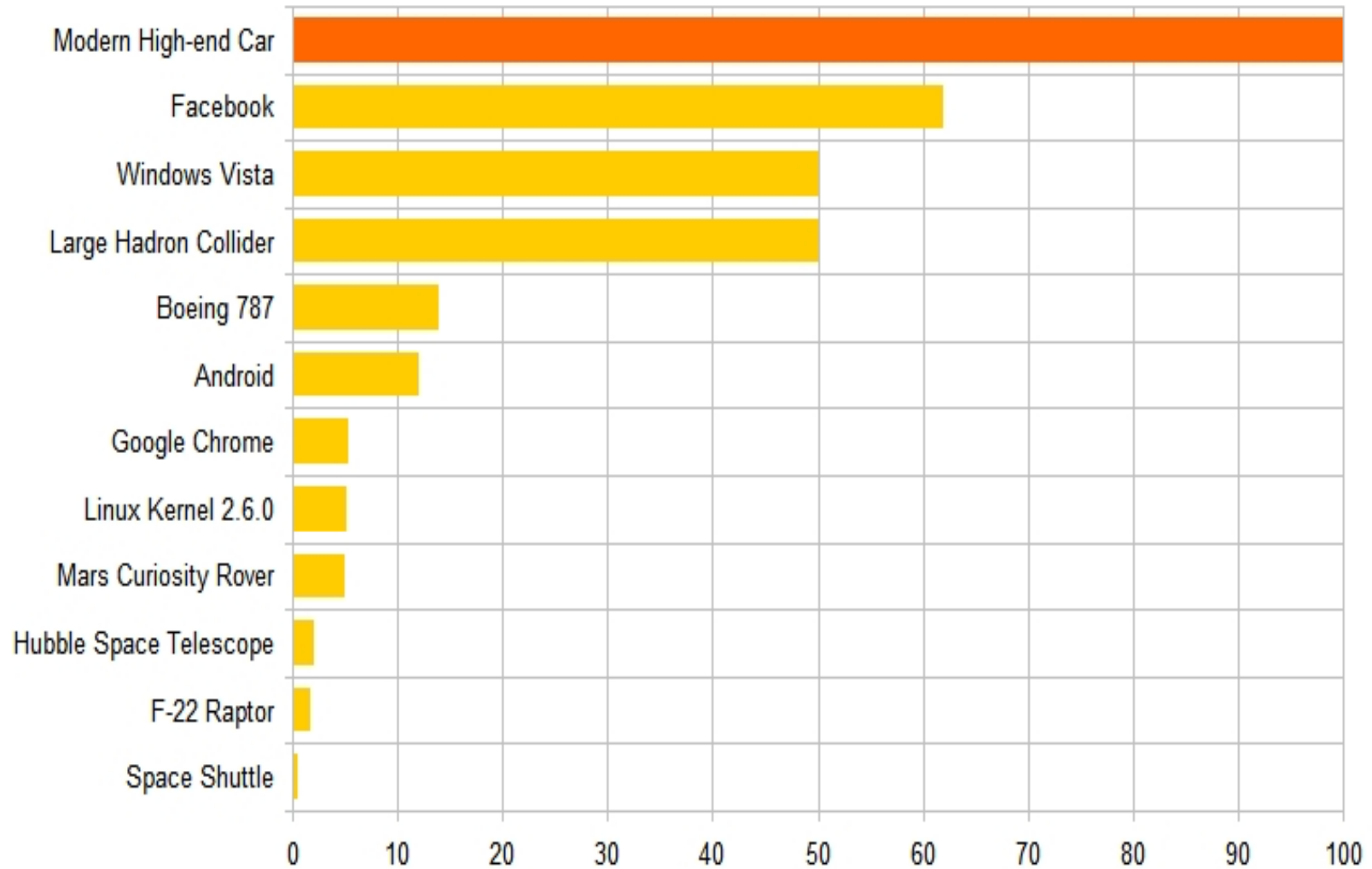
This course: CS 303 Operating Systems

- An operating system is an essential part of any computer system.
- **Examples of operating systems?**
- For desktops/laptops: Windows 10, OS-X, Linux.
- For mobile phones/tablets: Android, iOS.
 - From the past: Windows Mobile, Palm OS, Symbian.
- **What does an operating system do?**
- It lets us interact with the machine hardware.
- It provides us an environment in which we can run our programs.

Operating Systems (OS)

- This course is a core component of any Computer Science curriculum.
- Typically, there is enough material in OS to carve out two courses.
 - One at the under graduate level.
 - The other at the M. Tech./PhD level.
- An operating system is very detailed and sophisticated, consisting of millions of lines of code!
 - As we will see in the next slide.

Software Size (million Lines of Code)



Source: LinkedIn

Why study Operating Systems?

- **Some of you will actually design and build operating systems or their components.**
 - Who wants to work at Microsoft, Apple?
- **Many of you will build applications that utilize the core concept of operating systems.**
 - Facebook or Twitter anyone?
 - Design a more efficient way to load photos from Facebook (need knowledge of disk I/O).
- **Many of you may design hardware.**
 - Designing a chip @ Intel.
 - Need knowledge of programs and devices, and how they interact with the hardware.

Text Book

- For this course, we will be using the following text book:
- **“Operating System Concepts” by Silberschatz, Galvin & Gagne, 9th edition, Wiley India.**
- The book has 23 chapters.
- The authors suggest that the under graduate (UG) course may cover at chapters 1 – 13.

Pre-requisites

- **CS 204 – Computer Architecture.** Why?
 - An operating system talks to the hardware. Hence, a knowledge of the hardware structure/components & their interaction is essential in understanding operating systems.
 - We will be referring to CPU, memory, disk etc. throughout the book.
- **CS 201 – Data Structures.** Why?
 - An operating system is a (large) piece of code.
 - Data structures, such as linked lists are commonly used in implementing operating system code.
 - How is a process/program implemented in code?
- **Knowledge of C Programming Language.**
 - Used to implement operating system code.
 - C has been covered in the Intro to Programming course in the 1st year.

OS Book

- The book covers the operating systems concepts using easy to understand descriptions.
- Important operating system research results are covered at a basic level.
- For more details on the results, a list of papers is provided for “further reading”.
- The concepts and algorithms covered in the book are based on those used in existing operating systems.
- Aim is to present the concept in a general setting without tying to a particular operating system.
 - Although, examples from Linux & Windows are given in the chapters.
 - E.g. this is how Linux schedules programs, this is how Windows does memory management.

Syllabus

- Now, let us spend some time to go over the course syllabus.
- We will talk about the following:
 - The goals of the course.
 - Course contents.
 - Grading scheme.
 - Course policies.

End of Chapter 0