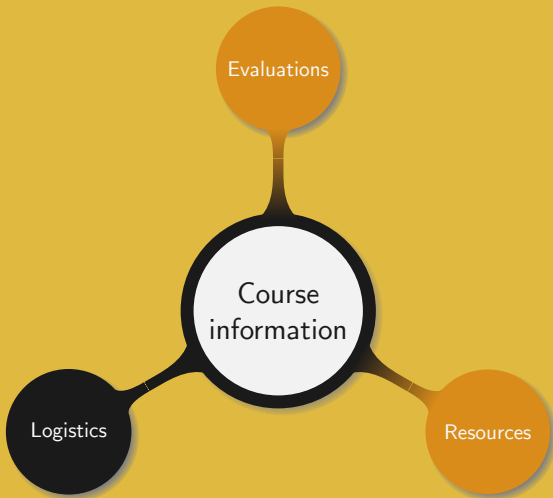# Introduction to Operating Systems

0. Course information

Manuel – Fall 2021

Teaching team:

- Instructor: Manuel (charlem@sjtu.edu.cn)

- Teaching assistants:
  - Yiding (colossuschang@sjtu.edu.cn)
  - Yuchi (citrate@sjtu.edu.cn)
  - Boming (bomingzh@sjtu.edu.cn)

Teaching team:

- Instructor: Manuel (charlem@sjtu.edu.cn)

- Teaching assistants:
    - Yiding (colossuschang@sjtu.edu.cn)
    - Yuchi (citrate@sjtu.edu.cn)
    - Boming (bomingzh@sjtu.edu.cn)

Important rules:

- When contacting a TA for an important matter, CC the instructor
- Prepend [VE482] to the subject, e.g. Subject: [VE482] Grades
- Use SJTU jBox service to share large files (> 2 MB)

Teaching team:

- Instructor: Manuel (charlem@sjtu.edu.cn)

- Teaching assistants:
    - Yiding (colossuschang@sjtu.edu.cn)
    - Yuchi (citrate@sjtu.edu.cn)
    - Boming (bomingzh@sjtu.edu.cn)

Important rules:

- When contacting a TA for an important matter, `CC` the instructor
- Prepend [VE482] to the subject, e.g. `Subject: [VE482] Grades`
- Use SJTU jBox service to share large files (> 2 MB)

Never send large files by email

Course arrangements:

- Lectures:
    - Tuesday 10:00 – 11:40
    - Thursday 10:00 – 11:40

- Labs:
    - Tuesday 18:20 – 20:55
    - Friday 12:10 – 14:45

- Manuel's office hours: Appointment (TBD)

- TAs' office hours: TBA

Main goals of this course:

- Understand the functioning of operating systems

- Become familiar with the internal structure of operating systems

- Be able to perform basic operating system coding

Main goals of this course:

- Understand the functioning of operating systems

- Become familiar with the internal structure of operating systems

- Be able to perform basic operating system coding

 *Be able to share in the development of an operating system*

Learning strategy:

- Course side:
  1. Understand how to efficiently use the CPU
  2. Know how to handle Memory, Input/Output, and Filesystems
  3. Get a basic idea of security and distributed systems

Learning strategy:

- Course side:
  1. Understand how to efficiently use the CPU
  2. Know how to handle Memory, Input/Output, and Filesystems
  3. Get a basic idea of security and distributed systems

- Personal side:
  1. Read and write code
  2. Relate known strategies to new problems
  3. Perform extra research

Detailed goals:

- Understand the general organisation of an OS
- Understand the hardware organisation
- Be familiar with the concept of process and threads
- Be able to solve common problems related to inter-process communication
- Be able to implement the most common scheduling algorithms
- Be able to analyse, prevent or solve deadlock issues
- Be familiar with the memory management and filesystems
- Be proficient at using Unix systems, spot particular parts of the kernel code, and write clean and well shaped code
- Understand the concept of security in an OS
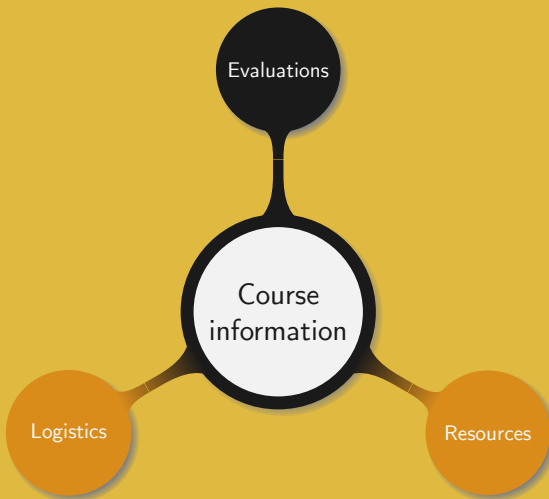
Homework:

- Total: 8
- Content: basic concepts, programming, scripting

Labs:

- Total: 9 + 3
- Content: improve programming skills, work on projects

Projects:

- Total: 3
- Content: shell, thread communication, scheduling

Extra: Linux kernel challenges

Grade weighting:

- Homework: 10%

- Projects: 40%

- Labs: 10%

- Midterm exam: 20%

- Final exam: 20%

Grade weighting:

- Homework: 10%
- Projects: 40%
- Labs: 10%

- Midterm exam: 20%
- Final exam: 20%

Assignment submissions:

- Late submission: -10% per day, not accepted after three days
- Dirty or hard to decipher: up to -10%

  Grades will be curved with the median in the range $[\![B, B+]\!]$

General rules:

- Not allowed:
  - Reuse the code or work from other students
  - Reuse the code or work from the internet
  - Give too many details on how to solve an exercise

General rules:

- Not allowed:
    - Reuse the code or work from other students
    - Reuse the code or work from the internet
    - Give too many details on how to solve an exercise

- Allowed:
    - Share ideas and understandings on the course
    - Provide general directions on where or how to find information

Documents allowed during the exams: none

Group works:

- Every student in a group is responsible for his group submission

- If a student breaks the Honor Code, the whole group is sent to Honour Council

Contact us as early as possible when:

- Facing special circumstances (e.g. full time work, illness...)

- Feeling late in the course

- Feeling to work hard without any result

Any late request will be rejected
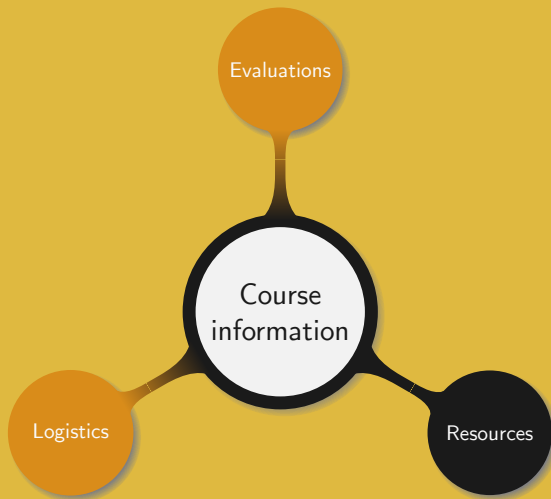
On Canvas platform:

- Course materials:
    - Syllabus
    - Lecture slides
    - Homework
    - Labs
    - Projects
    - Challenges

- Course information:
    - Announcements
    - Notifications
    - Grades
    - Surveys

Useful places where to find information:

- *Modern Operating Systems*, A. Tanembaum

- *Operating System Concepts*, A. Silberschatz

- *Operating Systems: Three Easy Pieces*, R. and A. Arpcaci-Dusseau

- OS creation: `http://wiki.osdev.org/Main_Page`

- Piazza

- Search information online, i.e. $\{$*internet* $\setminus \{$*non-English websites*$\}\}$

- Work regularly, do not wait the last minute

- Respect the Honor Code

- Go beyond what is taught

- Do not learn, understand

- Keep in touch with us

- Advice and suggestions are always much appreciated

Thank you!