



Introduction to OS Labs

Operating Systems



Learning by Doing-Advanced Track

- A full picture of design and implementation of operating systems
- Topics include Interrupts, system calls, context switches, virtual memory, synchronization, file systems
- Porting Linux 0.11 to ARM64 or RISC-V

Lab Overview-Advanced Track

Lab	DESCRIPTION	SCORE
0	Be familiar with Linux 0.11 source code, QEMU and debugging.	10%
1	Port the kernel to ARM64 or RISC-V, and say "hello world".	10%
2	Enable timers and interrupts, and print clock ticks.	30%
3	Realize a new scheduling algorithm and handle context switches.	20%
4	Implement sophisticated virtual memory features and show the performance.	30%
5	Extend page-fault mechanism based on Lab 4.	2pts
6	Manage 8G physical memory based on Lab 4&5.	2pts
7	Add more prevalent memory management features, say slab/slub/allocator.	2pts
8	Implement VFS (Virtual File System) and support MINIX, ext2 or 3 or 4.	2pts

Grading-Advanced Track

- You all are encouraged to finish all labs!
 - Finish Lab0-4, get full points of labs
 - Bonus Lab5-8, finish each one get 2 overall points
-
- Overall 100 points
 - Final Exam 50%
 - Homework 10%
 - Lab0-4 40%
 - Lab5-8 8 points (2 for each)

Grading-Advanced Track

- Overall ≤ 100 points
 - Maximum is 100
 - Final+HW+Lab0-4+Lab5-8 ≤ 100 points
- Examples
 - Final 90; Homework 100; Finished lab0-7
 - $90*50\%+100*10\%+100*40\%+6 = 101$ 100 overall points
 - Final 84; Homework 100; Finished lab0-8
 - $84*50\%+100*10\%+100*40\%+8 = 100$ overall points(总分)
- Final is hard, do bonus labs

Team-Advanced Track

- Team work is welcome and up to a maximum of 3 students. We will choose 3 to 5 groups for the labs.
- Apply today, contact TA
- Interview before Saturday, Sep 21th
- Feel free to ask and answer questions on QQ group

Deliverables-Advanced Track

- Per-team
- Source code
- Detailed lab report
 - with screenshots
 - describe what you have done and learnt
 - Feedback
- c.zju.edu.cn
 - Labs(projects), homework
 - Submission
- Be sure on time! A 10% penalty will be applied for each day of late submission.



Learning by Doing-Regular Track

- Learning important OS concepts by coding
- Topics include system calls, memory management, file systems
- Using latest Ubuntu and corresponding kernel
- Running on x86_64

Lab Overview-Reguar Track

Lab	DESCRIPTION	SCORE
0	Compiling kernel, add syscall, kernel module, traversal process and thread	20%
1	Memory management	50%
2	Encryption File System	30%

Grading-Regular Track

- Overall 100 points
 - Final Exam 50%
 - Homework 10%
 - Survey/Presentation 12%
 - Class Quiz 7%
 - Lab0-2 21%
 - No bonus

Deliverables-Regular Track

- Individual
- Source code
- Detailed lab report
 - with screenshots
 - describe what you have done and learnt
 - Feedback
- c.zju.edu.cn
 - Labs(projects), homework
 - Submission
- Be sure on time! A 10% penalty will be applied for each day of late submission.

Takeaway

- Two tracks
 - Choose according to your interest
 - Choose according to your capability
 - Can improve
- Self-learning
 - Google is your best friend, not Baidu
 - Search/Re-search → Find documents → Understand → Design → Implement → Test; all by yourself/your team
- Ask for help
 - Still can not solve, ask TA



Questions