

xv6: a simple, Unix-like teaching operating system

Russ Cox

Frans Kaashoek

Robert Morris

August 31, 2020

Contents

1	Operating system interfaces	9
1.1	Processes and memory	10
1.2	I/O and File descriptors	13
1.3	Pipes	15
1.4	File system	17
1.5	Real world	19
1.6	Exercises	20
2	Operating system organization	21
2.1	Abstracting physical resources	22
2.2	User mode, supervisor mode, and system calls	22
2.3	Kernel organization	23
2.4	Code: xv6 organization	24
2.5	Process overview	24
2.6	Code: starting xv6 and the first process	27
2.7	Real world	28
2.8	Exercises	28
3	Page tables	29
3.1	Paging hardware	29
3.2	Kernel address space	31
3.3	Code: creating an address space	33
3.4	Physical memory allocation	34
3.5	Code: Physical memory allocator	34
3.6	Process address space	35
3.7	Code: sbrk	36
3.8	Code: exec	37
3.9	Real world	38
3.10	Exercises	39
4	Traps and system calls	41
4.1	RISC-V trap machinery	42
4.2	Traps from user space	43

4.3	Code: Calling system calls	44
4.4	Code: System call arguments	45
4.5	Traps from kernel space	46
4.6	Page-fault exceptions	46
4.7	Real world	48
4.8	Exercises	48
5	Interrupts and device drivers	49
5.1	Code: Console input	49
5.2	Code: Console output	50
5.3	Concurrency in drivers	51
5.4	Timer interrupts	51
5.5	Real world	52
5.6	Exercises	53
6	Locking	55
6.1	Race conditions	56
6.2	Code: Locks	58
6.3	Code: Using locks	60
6.4	Deadlock and lock ordering	60
6.5	Locks and interrupt handlers	62
6.6	Instruction and memory ordering	62
6.7	Sleep locks	63
6.8	Real world	64
6.9	Exercises	64
7	Scheduling	67
7.1	Multiplexing	67
7.2	Code: Context switching	68
7.3	Code: Scheduling	69
7.4	Code: mycpu and myproc	70
7.5	Sleep and wakeup	71
7.6	Code: Sleep and wakeup	74
7.7	Code: Pipes	75
7.8	Code: Wait, exit, and kill	76
7.9	Real world	77
7.10	Exercises	79
8	File system	81
8.1	Overview	81
8.2	Buffer cache layer	82
8.3	Code: Buffer cache	83
8.4	Logging layer	84

8.5	Log design	85
8.6	Code: logging	86
8.7	Code: Block allocator	87
8.8	Inode layer	87
8.9	Code: Inodes	89
8.10	Code: Inode content	90
8.11	Code: directory layer	91
8.12	Code: Path names	92
8.13	File descriptor layer	93
8.14	Code: System calls	94
8.15	Real world	95
8.16	Exercises	96
9	Concurrency revisited	99
9.1	Locking patterns	99
9.2	Lock-like patterns	100
9.3	No locks at all	100
9.4	Parallelism	101
9.5	Exercises	102
10	Summary	103