

1 Topics:  
2  
3 Chapter 1-6, 8  
4  
5 OS design goals  
6  
7 GNU vs Linux  
8  
9 Dual Mode  
10  
11 OS Services  
12  
13 OS Structure  
14     ASM vs C in the kernel  
15     monolithic, microkernel, modular  
16  
17 Processes  
18     PCB  
19     What is in a process? Memory regions?  
20     How is an ELF related to a process?  
21  
22     What states can a process be in (for scheduling)?  
23  
24     What happens in a context switch?  
25     fork()/exec() (How does Fork/exec interact with the PCB?)  
26     dup2() (How does dup2 interact with the PCB?)  
27     pipe()  
28     shared memory  
29  
30     How does your CS460\_Shell work?  
31  
32 Threads  
33     how is a thread and process different? The same?  
34  
35     Benefits?  
36     Risks?  
37     Define thread-safe  
38  
39     thread models: 1to1, many to 1, many to many  
40     mutexes  
41  
42 Synchronization  
43     race condition/critical section  
44     3 properties of synchronization  
45  
46     why is hardware support necessary for synchronization?  
47  
48     dangers of synchronization  
49  
50 Scheduling  
51     what are the measurable goals of a scheduler?  
52     FCFS, SJF, SFJ (preemptive), RR

53 How do you predict the length of the next CPU burst?  
54 Multilevel queue scheduling  
55  
56 Kernel Modules  
57 loadable modules  
58 character device drivers  
59 structure, operation, ELF file  
60  
61 Memory  
62 memory heirarchy  
63 what is a cpu stall?  
64 logical vs physical address  
65 what is the memory management unit?  
66 first fit/best fit/worst fit  
67 fragmentation  
68 how does a function in a dynamically loaded shared library get called?  
69  
70  
71  
72  
73