Final Review: OS Class

Chee Yap

December 13, 2006

1 Introduction

In this quick review, we focus on material since the midterm. But up to 20% of the points may pertain to the pre-midterm material.

2 Pre-Midterm

1. We sometimes characterize an OS as an "extended machine". Give a detailed illustration of this characterization. Write 2-4 sentences to explain.

   A: For instance, most OS provides a general read(file-desc, start, num-bytes). This command is device independent, but depending the the devices, specific device drivers must be called, etc. Also, this command may involve several book-keeping steps, all hidden from users. This command thus provide an extended hardware service.

2. Name the three main architectural elements that an OS try to manage.

   A: CPU, RAM, Disk.

3. What exactly are we interrupting in an "interrupt"? Why do we need them? There are different kinds of interrupts, and often they have specific names. Name them and describe their functions.

   A: Normally, "interrupts" interrupts the CPU. We need them because software and hardware are running asynchronously, and they need to communicate. These are some of the kinds of interrupts:

   TRAP: a software-generated interrupt caused by user or error

   System Calls: application programs use these to access OS or kernel service.

   Exception: generated by error conditions

   I/O Interrupts: generated by I/O devices to communicate to the CPU.

4. What are signals?

   A: These can be viewed as interrupts between processes.

5. What is the Memory Hierarchy, and describe its main characteristics.

   A: The hierarchy refers to a sequence of memory devices, typically: register, cache, RAM, disk, tape. The main characteristic of this sequence is that the speed of the devices decreases as its capacity increases. Thus register (1 μs/100 B), cache (10 μs/1 MB), RAM (1 ms/1 GB), disk (100 ms/1 TB), tape (?/?).

6. Besides the speed/size difference, there is a major difference between memory and disk with regards to how they are accessed by the CPU.

   A: Memory reference is automatically handled by hardware while disk reference requires an explicit request by the OS.

7. Describe the mechanics of a system call.

   A: Sibers. p.30,32.

8. What is the relation between a process and a program?

   A: A process is a program in execution.

9. Name the three main OS functions in managing multiprocessing.

   A:
   – Keeping track of processes (maintaining the process table: create/delete)
   – Scheduling processes
   – Synchronization of processes
   – File management
   – Memory management
   – OTHERS (networking, protection, security, windowing, I/O)

10. What are the two simplest scheduling algorithms? What are some things left to be desired?

    A: FCFS and RR.

11. To schedule processes optimally, we want to classify processes into two types. What are they, and how do we use this classification in scheduling?

    A: CPU-bound and I/O-bound processes. We should give priority to I/O-bound processes.

12. What are scheduling events?

    A: there are 4 but the last two of them pertain only to preemptive scheduling:

    Process terminates, Process blocks, Process move from blocked to ready, Process move from running to ready.

13. What are some criteria for evaluating schedulers?

    A: CPU Utilization, throughput, wait time, turn around time, fairness, balance, predictability.

14. In what sense and under what conditions is SJF optimal?

    A: Optimality Sense: minimum wait time. Condition: all jobs are released at the same time.

    Can the condition be removed? Yes, suppose we can preempt. Then each time a job joins the queue, we may preempt the current job with the new job if it has a shorter time to completion.

3 Deadlocks

1. What is a deadlock? Give an example.

   A: No process can make progress.

   E.g., dining philosopher.

2. List the sufficient and necessary conditions for a deadlock

   A: mutex, hold-and-wait, non-preemption, circularity

3. Describe Holt's Resource graph model, and how deadlock is represented. What is the limitation of this model?

   A: bipartite directed graph. $P \rightarrow R$ means prcess P is waiting on resource R $R \rightarrow P$ means resource R is held by process P

   Limitation: only 1 copy of each resource.

   REMARK: this a run-time graph.

4. Describe a program involving two processes P, Q that can cause a deadlock. Do the run-time illustration by Holt's graph to illustrate deadlock.

   A:

   P: repeat(request R; request S; Release R; Releast S;)

   Q: repeat(request S; request R; Release R; Releast S;)

5. Describe the general resource-process model, and describe the parameters in such a model.

   A:
   – n processes (p1,..,pn)
   – m resources (r1,..,rm)
   – num resources E=(e1,..,em)
   – max request matrix (nxm) M(i,j)
   – (pending) Request matrix (nxm) R(i,j)
   – (current) Allocation matrix (nxm) A(i,j)
   – free vector F=(f1,...,fm) $\leq$ E
   – state = (A,R) where A+R $\leq$ M
   – safe state: we can allocate such that each process terminates

6. Describe the Banker's algorithm:

   A:

4 Unix Questions

1. T/F: there is one i-node per file. Explain.

   A: True in the sense that each physical file has exactly one i-node. If we also view directories as files, then every i-node also corresponds to a file.

   It is not strictly true if we consider links (now, multiple copies of the files have only one i-node). Also, since I/O devices have i-nodes, but strictly speaking they are not regular files.

2. How can unix treat I/O devices as files?

   A: Just like files, we first need to open an I/O device for reading or writing. This gives us a file-descriptor (fd). We henceforth use this fd to read and write to the device.

3. Describe an i-node.

   When we open a file, its i-node is kept in main memory.