Operating Systems, Evaluation 3

CS30002, Spring 2021

12:15 noon to 1:25 pm, 14th April, 2021
Full marks: 50
Answer ALL questions

IMPORTANT INSTRUCTIONS

Taking the exam: You need to log into Google meet (personalized links will be provided close to the exam date), keep your video on during taking the test (so that we can monitor you during the exam). **YOU HAVE TO USE PEN AND PAPER TO GIVE THE EXAM.**

Decorum: Throughout the examination, you are strictly expected to have your cameras on, directing towards your work-space including yourself. Arrange your laptops/desktops/mobiles beforehand to save time during the examination. Disconnecting video for a long duration will be grounds for suspecting malpractice.

You need to keep your workplace and your hands visible to us. However, avoid the visibility of your answers to the rest of students. Once you open your question paper, refrain yourself from using your PC/laptop/mobile for searching for anything or typing during the exam.

Submission: You can do either of two things -- (i) take pictures of your answer script pages, name the pictures page1.jpg, page2.jpg, page3.jpg, etc., zip the pictures and upload the zipped file via CSE Moodle, or (ii) Put all the pages sequentially in a pdf file and upload the pdf to CSE Moodle. Note that the scheduled time includes both writing time and time needed to upload your answer-script.

Submissions must be through the course Moodle, by 1:25 PM (according to the Moodle clock). The actual exam will be from 12:15 to 1:15 PM and you have a dedicated 10 minutes time to upload the exam to moodle.

If you miss this deadline for Moodle submission, then you need to email your submission to TA Anju Punuru (anjupunuru@gmail.com) within 1:35 PM; there will be a penalty of 10 marks for such late submissions. No submission will be allowed after 1:35 PM. If any submission reaches Anju's mailbox after 1:35 PM (according to the timestamp of the mail), it will be rejected.

Tip: Install Adobe scan on your mobile phone to make the whole process easier. In that case, your laptop acts as a camera, while you are using your mobile for checking the questions, scanning and uploading the answers. [This is only a suggestion that can make things simpler for you, not a necessity.]

Policies: Note that, if we face problems with your answer script, e.g., cannot open your submitted zipped file, cannot read the text in pictures (due to bad resolution), cannot determine the page order from the file names (or if the pages in the pdf are jumbled up), it will affect your marks.

Malpractice: If any group of students is found to have similar answers/working in their answer sheets, ALL of them will receive the maximum penalty with no grace. We will not distinguish between who supplied answers and who copied; everyone involved will receive the maximum penalty. We expect you to NOT take help from the internet, your copies, textbooks, slides or video recordings during the exam. Note that this is NOT an open-book exam. Also, you should NOT discuss answers with anyone during the scheduled exam time. If found otherwise, you will be penalized.

PLESE WRITE YOUR NAME AND ROLL NO. ON THE TOP OF THE FIRST PAGE OF YOUR ANSWER SCRIPT. WE WILL NOT EVALUATE YOUR ANSWER SCRIPT WITHOUT IT.

---

1. Answer the following. **[4 + 4 + 6 = 14]**

   a) Consider that the average size of a process is 8 Kbytes, and each page table entry is of size 4 bytes. What will be the optimum page size that minimizes the total memory overhead? Derive any expression you use.

   b) Explain the second-chance page replacement algorithm that uses *reference bits* with the help of an example. How can the performance of the method be improved by additionally considering the *modify bits*?

   c) Consider a small system where the virtual memory page size is 2K (2048 bytes), and main memory consists of 4 page frames. Now consider a process that requires 8 pages of storage. At some point during its execution, the page table is as shown below:

Virtual page	*Valid*	*Physical page*
0	No	
1	No	
2	Yes	1
3	No	
4	Yes	3
5	No	
6	Yes	0
7	Yes	2

      i. List the virtual address ranges (in hexadecimal) for each virtual page.

ii. Give the main memory (physical) addresses *in hexadecimal* for each of the following virtual addresses (given in decimal), mentioning which of these will result in page faults: (i) 8500, (ii) 14000, (iii) 5000, (iv) 2100.

2. Answer the following. **[4 + 4 = 8]**

a) In a computer system the page tables are stored in main memory that has an access time of 100 nsec (1 nsec = 10^{-9} sec). The TLB can hold 8 page table entries and has an access time of 10 nsec. During the execution of a program, it is found that 85% of the time the page being referenced is found in TLB, and only 2% of the memory references lead to page faults. The average time for page replacement is 2 micro sec (1 micro sec = 10^{-6} sec). Compute the average memory access times for (show your reasoning):

i. A single-level page table organization

ii. A three-level page table organization

b) Assume that you have a page-reference string for a process with *m* frames (initially all empty). The page reference string has length *p*, and *n* distinct page numbers occur in it.

i. What is a lower bound on the number of page faults?

ii. What is an upper bound on the number of page faults?

3. Answer the following **[3 + (2 + 2) + 3 = 10]**

a) Consider an x86 computer with the following memory architecture

- 32-bit virtual address space
- 32-bit physical address space
- 4 KiB page size
- Two-level page table
- 32-bit page table entry (PTE) size

How many bits of the 32 bit virtual address are used for (i) index of outer page table (ii) index of inner page table (iii) offset in the page. Why? (No marks without explanation)

b) For the following page reference string, estimate the number of page faults that would occur for (i) FIFO replacement, (ii) LRU replacement algorithms with four frames in memory: 7, 3, 5, 2, 1, 0, 7, 4, 7, 1, 2, 0, 2, 6, 3, 4, 1, 5, 2, 7. Show the calculation (No marks without calculation)

c) Derive an expression for the average memory access time in a demand paging memory management system that also includes a TLB. Make relevant assumptions.

4. Answer the following (no marks without calculation). **[4 x 2 = 8]**

a) Consider a Really Simple FileSystem, RSFS, where each inode has 10 direct pointers. each pointer can point to a single file block. Direct pointers are 32 bits in size (4 bytes). What is the maximum file size for RSFS? Show the calculation.

b) Consider a filesystem, based on extents, thus called EFS. An extent is a construct which can span multiple blocks. Extents have a pointer (base address) and a length (in blocks). Assume the length field is 8 bits (1 byte). Assuming that an inode contains exactly one extent. What is the maximum file size for EFS? Show the calculation.

c) Consider a filesystem that uses direct pointers, but also adds singly indirect pointers and double-indirect pointers. We call this filesystem, IFS. Specifically, an inode within IFS has 1 direct pointer, l indirect pointer, and 1 doubly-indirect pointer field. Pointers are 4 bytes (32 bits) in size. What is the maximum file size for IFS? Show the calculation.

d) Consider a file system called CFS, which tries to save as much space as possible within the inode. Thus, to point to files, it stores only a single 32-bit pointer to the first block of the file. However, blocks within CFS store 4,092 bytes of user data and a 32-bit next field (much like a linked list), and thus can point to a subsequent block (or to NULL, indicating there is no more data). What is the maximum file size for CFS (assuming no other restrictions on file sizes)? Show the calculation.

5. Imagine that you want to design a file system over a SSD for an iot device. The SSD has a block size of 4KB, and the latencies to read and write a block are 100 micro sec, and 200 micro sec, respectively (1 micro sec = 10^{-6} sec). In your file system built over this SSD, you decided to use the following inode structure: 15 direct pointers, a singly indirect pointer, a triply indirect pointer. Each pointer is 4 bytes. Now Answer the following **[3 + 3 + 4 = 10]**

   a) If an application *readdata.exe* using this SSD reads 256 byte random data chunks what is the rate of reading data for *readdata.exe* (i.e., how much data read per second by *readdata.exe*). Ignore block transfer time.

   b) What is the maximum file size that your file system can support? Show the calculation (no marks without calculation)

   c) How long does it take to write a 4MB file in your file system? Show the calculation (no marks without calculation)