Operating Systems, Final Evaluation

CS30002, Spring 2021

8:15 am to 9:45 am, 8th April, 2022
Full marks: 60
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. You should be visible on camera even while scanning/uploading your answer-script. 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.**

Submissions must be through the course Moodle, by 10:00 am (according to the Moodle clock). The actual exam will be from 8:15 to 9:45 am and you have a dedicated 15-minute slot to upload your answer-script to Moodle. If you miss this deadline for Moodle submission, you need to email your submission (with OS Final Test - <Roll No> as subject) to TA Soham Poddar (sohampoddar@kgpian.iitkgp.ac.in) within 10:15 am; there will be a penalty of 20 marks for such late submissions. No submission will be allowed after 10:15 am. If any submission reaches the TA's mailbox after 10:15 am (according to the timestamp of the email), it will be rejected.

You are allowed to submit via one medium only - either via Moodle (by 10:00 am) or via email to the TA (during 10:00 - 10:15 am). If we find your submission on Moodle, that is only what we will consider.

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 strictly penalized.

Tip: Install Adobe Scan app on your mobile phone to make the whole process easier. In that case, your laptop/PC 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. You are free to use other devices for scanning, e.g., if you have a scanner available. Note: scanning should also be done in front of the camera.]

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.

Question 1. **[2 x 10 = 20 marks]**

Please answer whether each statement is TRUE or FALSE with justification (no marks without justification)

a) Invalid page table entries (PTE) might be cached into the TLB.

b) On every page fault the OS either brings in a page from disk to memory or terminates the process that issued the memory access (if the page does not exist or the process does not have proper permission).

c) The clock algorithm will always evict the same pages as LRU.

d) In the FAT file system there is no *theoretical* limit on the maximum size of individual files (aside from the size of underlying storage device).

e) If two processes simultaneously hold open file descriptors corresponding to the same file, then two instances of an in-memory inode exist for that file.

f) A kernel interrupt handler runs in a thread.

g) The Swapped-Out states in a process state transition diagram are required to keep track of the processes that do not require CPU time.

h) One file system may contain a soft link to a file in a different file system.

i) One file system may contain a hard link to a file in a different file system.

j) The FAT file system supports hard links.

Question 2. **[6 x 1 = 6 marks]**

It is time for elections amidst the pandemic. Consider that a polling station has one polling agent, one voting machine, and a waiting room with exactly N seats (placed

maintaining social distancing). Voters come to this polling station to cast their votes following some rules. If a voter arrives, and finds that one voter is casting the vote and all N waiting room seats are occupied, he/she immediately leaves the polling station according to the rules. Otherwise, the voter sits on one of the empty waiting room seats and waits. If there is no one casting vote at a particular point of time, any one of the waiting voters can approach the agent and cast his/her vote. The agent goes to sleep if there is no voter in the polling station. If a voter comes and finds the agent sleeping, the voter wakes up the agent.

We want to simulate this situation considering that the agent and each voter is implemented as a thread/process, all of which will run concurrently. Assume that the following three semaphores and one variable are already declared and initialized as shown:

semaphore agentReady = 0, voterReady = 0, accessSeats = 1;
int numFreeSeats = N;

Every semaphore has P( ) (or, wait( )) and V( ) (or, signal( )) functions available as <semaphore>.P( ) and <semaphore>.V( ). E.g., the semaphore agentReady has the functions agentReady.P( ) and agentReady.V( ) available. Fill in the blanks in the functions below to implement the agent and a voter, so as to ensure that the agent and voters are synchronized and free from deadlock. **Note that each blank can have one or more P( ) or V( ) statements.**

```
void agent( ){
   while (true) {
         _ _ _ _ _ _ _ _ _ _ _ blank (i) _ _ _ _ _ _ _ _ _ _ _

         numFreeSeats = numFreeSeats + 1;    // one voter will cast vote;
   one seat freed

         _ _ _ _ _ _ _ _ _ _ _ blank (ii) _ _ _ _ _ _ _ _ _ _ _
         helpVoterCastVote( );

         _ _ _ _ _ _ _ _ _ _ _ blank (iii) _ _ _ _ _ _ _ _ _ _ _
   }     // end while( )
    }    // end agent( )

    void voter( ){

         _ _ _ _ _ _ _ _ _ _ _ blank (iv) _ _ _ _ _ _ _ _ _ _ _
         if ( numFreeSeats > 0 ){
               numFreeSeats = numFreeSeats - 1;    // sit on one free seat
               _ _ _ _ _ _ _ _ _ _ _ blank (v) _ _ _ _ _ _ _ _ _ _ _
               castVote( );
         } else{
               _ _ _ _ _ _ _ _ _ _ _ blank (vi) _ _ _ _ _ _ _ _ _ _ _
               leaveWithoutCastingVote( );
         }     // end else
   }     // end voter( )
```

Question 3 **[4 + 6 = 10 marks]**

3.1. N processes share M resources of the same type that can be reserved and released only one at a time. The maximum need of each process does not exceed M, and the sum of all maximum needs is less than M + N. Show that a deadlock cannot occur in the system.

3.2. A system has four processes and five allocatable resource types each having multiple instances. The current allocation and maximum needs are in the tables below. In these tables, each column shows the number of instances for a particular resource type (out of the five resource types), and the resource types are ordered in the same order in each table.

	Allocated	Maximum	Available
Process A:	1 0 2 1 1	1 1 2 1 3	0 0 *x* 1 1
Process B:	2 0 1 1 0	2 2 2 1 0	
Process C:	1 1 0 1 0	2 1 3 1 0	
Process D:	1 1 1 1 0	1 1 2 2 1	

Examine whether the state is safe for (i) $x = 1$, and (ii) $x = 5$. Show all calculations.

Question 4 **[6 + 6 = 12 marks]**

4.1. The hardware of a memory management system is equipped with an L1-cache (average miss-rate: 0.35, access-time: 1 clock) and L2-cache (average miss-rate: 0.2, access-time:5 clocks) and a main-memory (access-time: 100 clocks), a TLB (average miss-rate: 0.5, access-time: 1 clock).

Consider a virtual memory for this system using a two level hierarchical page table system and a 32-bit virtual address space. Any page table in this system has the same size (irrespective of the level), with each PTE of size 4 bytes and are stored in memory. The page size in this system is the same as that of the size of the page table. Page faults (average rate: 0.002, service time: 20000 clocks) are serviced in the usual interrupt based manner.

Assume only a single process is using the entire memory system (so cache, TLB etc. are not getting modified due to any other process) and the cache entries are physical address mapped until stated otherwise. The first level page table is always available. Make other suitable assumptions where necessary.

i. Assuming that the CPU has the physical address of a memory location, what is the average time for accessing a memory location, taking only the L1, L2 and the main memory into account?

ii. What is the page size being used?

iii. What is the average time for retrieving data when the CPU has only the virtual address at hand? Take the complete flow into account.

4.2. For the following page reference string, estimate the number of page faults that would occur for (i) FIFO (ii) LRU replacement algorithms with four frames in memory: 7, 3, 5, 2, 1, 0, 7, 1, 7, 1, 2, 0, 2, 6, 3, 4, 1, 5, 2, 7, 0, 7, 4, 7, 1, 2, 0. Show the calculation.

Question 5 **[6 + 6 = 12 marks]**

5.1. You designed a filesystem with the following inode structure with 4n byte block size: most are direct pointers except for one single, one double and one triple indirect pointer. The pointer size is 4 bytes. Now you implement your file system on your smart washing machine which has 10000 blocks for storage (aside from separate storage reserved for your inodes). If the OS of the washing machine only uses one file, show that even when the file reaches its theoretical maximum size in your system, it will either grow beyond the storage space (resulting in an error message) or there will be external fragmentation, i.e., some block will remain unused.

5.2. Now, you want to try your hand at building file system for a streaming service (essentially a YouTube clone). So you need to create a file system for such a streaming service which is used only for storing and playing videos for millions of users. Now answer the following questions (no marks without explanation)

i. What is the expected access pattern (random, sequential, somewhere in between) for your storage system. Why?

ii. What would be your preferred file allocation method? Explain your design decision.

iii. Would you recommend to use a SSD or a hard disk or some other type of storage for your file system keeping in mind the cost difference, access pattern and any other factors you can consider? Why?