UNIVERSITY
OF SKÖVDE

School of Informatics

WRITTEN EXAMINATION

Course Operating systems G1F, 6hp

Course Operating systems G1F, 7.5hp

Sub-course

Course code IT390G IT391G

Credits for written examination 6hp

Date 2023-12-07 Examination time 08:15-13:30

Examination responsible András Márki

Teachers concerned Simon Butler

Aid at the exam/appendices

Other

Instructions

- ☐ Take a new sheet of paper for each teacher.
- ☒ Take a new sheet of paper when starting a new question.
- ☒ Write only on one side of the paper.
- ☒ Write your name and personal ID No. on all pages you hand in.
- ☒ Use page numbering.
- ☒ Don´t use a red pen.
- ☒ Mark answered questions with a cross on the cover sheet.

Grade points

ECTS A: >=80% of points on the whole exam AND >= 25% on all of parts 1-2

ECTS B >=72.5% of points on the whole exam AND >= 25% on all of parts 1-2

ECTS C >=65% of points on the whole exam AND >= 25% on all of parts 1-2

ECTS D >= 57.5% of points on the whole exam AND >= 25% on all of parts 1-2

ECTS E >= 50% of points on the whole exam AND >= 25% on all of parts 1-2

ECTS F <50% of points on the whole exam OR <25% on any of parts 1-2 on their own

UNIVERSITY
OF SKÖVDE

Examination results should be made public within 18 working days

Good luck!

Total number of pages

Before you start: multiple-answer questions can have multiple correct answers. To get full marks, you should have selected all correct answers, and you should not have selected any wrong answer; **each correctly completed / not completed alternative is rewarded with 0,2 marks**. Enter the answer to the first part of the exam directly into the exam paper. Answers given otherwise do not count. Select an option by drawing a cross in the box. If you change your mind, fill in the whole box.

Properly completed responses: ☒

Amended response: ■

For the parts where we expect a computation, please make sure you answered all questions and do have your computations present. Computations are preferably answered on the writing sheets. For the figures, please make sure that all necessary parts are present.

As you are answering on the sheets, you must write your name and (Swedish) person number even on the exam sheets.

UNIVERSITY
OF SKÖVDE

Name: ______________________________ Person number: __________________

Part 1 (MARD):

- Describe and compare terms, techniques, and algorithms in process and thread handling, for example scheduling, communication, synchronization, and deadlock handling,

1 Process concepts (10 Points)

1.1 What characterizes an CPU-bound process? (1 Point)

☐	Common for some computation-heavy applications.
☐	Uses the I/O mostly via interrupts run on the kernel.
☐	Usually requires high memory bandwidth.
☐	Is usually not dependent of the system disk.
☐	It is bottlenecked by the bookkeeping module, as CPU is an abbreviation for Central Procedure Dispatch in this case.

1.2 Which of the following registers are necessary for communicating between the CPU and the memory? (1 Point)

☐	Memory buffer register (MBR)
☐	Instruction register (IR)
☐	Memory address register (MAR)
☐	Random access register (RAR)
☐	Bearing Access Register (BAR)

UNIVERSITY
OF SKÖVDE

Name: ____________________________ Person number: __________________

1.3 Compare the **size** and the **speed** of the following devices in the storage-device hierarchy: hard-disc drive, solid-state drive, main memory, magnetic tape. You can draw a figure if needed. (2p)

1.4 Give an example of when it is sufficient to only have a short-time scheduler in the operating system, and give another example when having a long-term scheduler is also preferred. Give a short motivation for both examples (2p).

UNIVERSITY
OF SKÖVDE

Name: ______________________________ Person number: __________________

1.5 Draw a diagram representing which states a process can have (4p). Make sure that you include the state transitions too! (You can draw this on an extra sheet)

UNIVERSITY
OF SKÖVDE

Name: ______________________________ Person number: ____________________

2 Threads (Total: 10 Points)

2.1 What are the benefits of using threads? (1 Point)

☐	Deadlock avoidance becomes easy
☐	Scalability, which is important for utilizing modern CPU architectures
☐	Execution order is trivial when using multiple threads
☐	Resource sharing within a process
☐	Modularity, as different responsibilities within a process can be divided easier

2.2 Which of the following are difficulties when utilizing multiple threads? (1 Point)

☐	Balancing workload between threads
☐	Data dependencies between threads
☐	Dividing workload between threads
☐	Storing and splitting the data between threads
☐	Verification and validation

2.3 What is true about thread cancellation? (1 Point)

☐	Deferred cancellation terminates the target thread immediately
☐	Asynchronous cancellation checks the target thread periodically if it should be cancelled
☐	Thread cancellation occurs only when the thread is finished with its execution
☐	Thread cancellation is enabled by default.
☐	A thread can disable its cancellation.

UNIVERSITY
OF SKÖVDE

Name: ____________________________ Person number: __________________

2.4 What is true about Amdahl's law? (1 Point)

☐	The more parallel parts a program has, the more it will scale with more processor cores.
☐	The expected speedup is smaller or equivalent to 1 / (serial_portion + (1-serial_portion)/number_of_processing_cores).
☐	Doubling the number of processors can yield up to 100% speedup according to the law.
☐	Independent of how your program works, there will be no speedup when increasing the number of processors over 16.
☐	Amdahl's law gives an upper limit on how much speedup we can expect of program using multiple processors when more processors are added to the system.

2.5 What is true about PThreads? (1 Point)

☐	Is short for pointer* threads
☐	Must be always provided as kernel-level
☐	Concerns the actual implementation of threads and not only the specification
☐	Commonly used
☐	Must be always provided as user-level

UNIVERSITY
OF SKÖVDE

Name: ____________________________ Person number: __________________

2.6 Draw a comparison between a single-threaded and multithreaded process, displaying the inner structure of both. Make sure to label the necessary parts accordingly. (You can draw this on an extra sheet) (3 Points).

2.7 Draw the many-to-many multithreading model. Make sure to label the necessary parts accordingly. (You can draw this on an extra sheet) (2 Points)

UNIVERSITY
OF SKÖVDE

Name: ______________________________ Person number: ____________________

3 Process scheduling (10 Points)

3.1 What is true about scheduling? (1 Point)

☐	With non-preemptive scheduling, the CPU can be taken away from a process without the cooperation of the given process.
☐	Long-term scheduler selects from among the processes in the job queue.
☐	The ready queue is managed by short-term scheduling.
☐	CPU scheduling only happens when a process terminates.
☐	Both preemptive and non-preemptive scheduling can be present in the same OS.

3.2 What is true about the scheduling goals of real-time processes? (1 Point)

☐	Minimize average response time
☐	Complete process by given deadline
☐	The solution should scale
☐	CPU utilization should be maximal
☐	The solution should be fair amongst processes

3.3 What is true about the evaluation of scheduling algorithms? (1 Point)

☐	Simulations cannot be done on a whiteboard
☐	Implementing a scheduling algorithm within a real system is usually the most expensive
☐	Neurooptimistic evaluation gives results for every possible workload input
☐	Analytical evaluations are not using mathematical models
☐	Deterministic modeling can be time-consuming for a project

UNIVERSITY
OF SKÖVDE

Name: ____________________________ Person number: ____________________

3.4 What is true about homogenous multiprocessing? (1 Point)

☐	The instruction set of the CPU cores must be the same within the CPU
☐	CPU scheduling is easier compared to non-homogenous multiprocessing
☐	All CPU cores are physically alike within the CPU
☐	It is a common solution on servers where multicore performance and load efficiency is important
☐	It is an uncommon concept that is only important for legacy systems as it makes software development harder

UNIVERSITY
OF SKÖVDE

Name: ______________________________ Person number: __________________

3.5 Scheduling algorithms: Assume the following processes arrive for execution at the indicated time with the specified priority and the length of their CPU-burst time given in milliseconds. (You can give your answer on the extra sheet)

Process	Burst time (ms)	Priority	Arrival time (ms)
P1	2	1	0
P2	2	2	0
P3	4	3	2
P4	2	4	2
P5	2	2	3

3.5.1 Give a Gantt chart illustrating the execution of these processes using FCFS, Round Robin (quantum=3), and Priority (Non-pre-emptive). (3 Points)

3.5.2 Calculate the average waiting time for each of the above scheduling algorithms. (3 Points)

UNIVERSITY
OF SKÖVDE

Name: ____________________________Person number: __________________

4 Synchronization (10 Points)

4.1 If you have two threads modifying the same variable int var = 2 in parallel, with their respective code being (without synchronization) (1 Point):

Thread A	**Thread B**
var=var+2	**var=var-2**

☐	It is possible to get var = 4 as a result
☐	It is possible to get var = 0 as a result
☐	It is possible to get var = 2 as a result.
☐	It is possible to get var = -2 as a result
☐	The result of the variable var can be different after each execution of the above code

4.2 Which of the following criteria should a proper solution to the critical-section problem fulfill? (1 Point)

☐	Bounded waiting
☐	Circular queue
☐	Mutual exclusion
☐	Progress
☐	Excitement

UNIVERSITY
OF SKÖVDE

Name: ______________________________Person number: __________________

4.3 What is true about synchronization hardware? (1 Point)

☐	Disabling interrupts is an inefficient solution on modern hardware for synchronization and is usually avoided
☐	It is a thing of the past
☐	Modern machines still need to provide atomic hardware instructions
☐	Synchronization hardware uses locks
☐	Disabling interrupts is unappropriate for multiprocessor systems.

4.4 What is true about the bounded-buffer classical problem? (1 Point)

☐	It is sometimes called the producer-consumer problem.
☐	It has two kinds of processes: a consumer and a producer.
☐	Uses three types of semaphores: Mutex, full, and empty.
☐	Uses a buffer
☐	Regarded as a classical problem where the cooperating processes must follow each other in strict order.

4.5 What is true about the readers-writers classical problem? (1 Point)

☐	It represents a problem where processes need to access multiple resources at the same time to perform their goal.
☐	Writers can both read and write the data
☐	Depending on the implemented solution, starvation is possibly an issue.
☐	Has a variant prioritizing readers
☐	Has a variant prioritizing writers

UNIVERSITY OF SKÖVDE

Name: ____________________________ Person number: __________________

4.6 What is true about the dining philosophers classical problem? (1 Point)

☐	Is a form of selective mutual exclusion, as some processes (i.e., readers) can be in a critical section simultaneously, but not others.
☐	The solution only works with exactly five philosophers.
☐	Deadlocks can be handled with asymmetric solutions depending on the number of the philosopher.
☐	Philosophers in the problem are either eating or thinking
☐	There is a variant focusing on (brief) historical time ordering, called Proust's fika.

4.7 What happens if you use a semaphore incorrectly as below? Motivate your answer. (2 Point)

1) Signal (mutex) ... signal (mutex)
2) Wait (mutex) ... wait (mutex).

UNIVERSITY
OF SKÖVDE

Name: ____________________________ Person number: __________________

4.8 Draw the general inner structure of a process containing a critical section. Make sure to label the necessary parts accordingly. (2 Points)

UNIVERSITY
OF SKÖVDE

Name: ____________________________ Person number: ________________

5 Deadlocks (10 Points)

5.1 What are the common methods for handling deadlocks? (1 Point)

☐	Ignore that the problem exists
☐	Allow the system to recover from deadlocks
☐	Adding a NAS to the system
☐	Ensure that the system will never enter a deadlock
☐	Using a homogenous CPU architecture

5.2 Which of the following conditions must hold simultaneously for a deadlock to arise? (1 Point)

☐	Central Finite Curve
☐	No preemption
☐	Mutual exclusion
☐	Hold and wait
☐	Circular wait

5.3 When recovering from a deadlock and aborting one process at a time (selecting them as victims), which of the following factors can be considered to choose a potential process as a victim? (1 Point)

☐	Resources needed by the process to complete
☐	Type of process (interactive or batch)
☐	How many victims are needed
☐	How many users are logged in
☐	Priority of the process

UNIVERSITY
OF SKÖVDE

Name: ______________________________ Person number: __________________

5.4 Draw an example of a resource allocation graph (3p)

5.4.1 Containing a cycle and a deadlock

5.4.2 Containing a cycle but not a deadlock

Make sure to use the correct notation and label the necessary parts accordingly. (You can give your answer on the extra sheet)

5.5 Apply the banker's algorithm to the example below and determine if a safe sequence exists. The total amount the resource are: A=5, B=7, C=6. Write down the intermediate results for each step. (4 Points) (You can give your answer on the extra sheet)

	Allocation			Max		
	A	B	C	A	B	C
P0	0	0	0	0	1	0
P1	3	2	2	4	3	3
P2	1	3	1	5	4	5
P3	1	1	0	1	1	1
P4	0	1	2	4	4	4

UNIVERSITY
OF SKÖVDE

Name: ____________________________ Person number: ________________

Part 2 (BUTS):

- describe and compare terms, techniques, and algorithms in primary memory management, for example addressing, address binding, paging, segmentation, and virtual memory;
- describe and compare terms, techniques, and algorithms in secondary memory management, for example file allocation and scheduling of disc operations;
- discuss the problems of protection and security in modern operating systems

6 Memory management (10 Points)

6.1 Which of the following statements about memory management are true? (1 point)

☐	The physical address space used by a process are defined by a pair of base and limit registers.
☐	Compile time address binding allows the operating system to move processes in physical memory.
☐	Pages in virtual memory are the same size as frames in physical memory.
☐	Re-entrant or read-only code is used by shared processes.
☐	User mode processes (programs) can not adjust the base and limit registers defining the space the process occupied in memory.

6.2 Which of the following statements about memory management are true? (1 point)

☐	Searches (look-ups) in a translation look aside buffer (TLB) depend on the number of entries in the TLB and have a computational complexity of O(n).
☐	Logical (virtual) and physical addresses are the same at execution time.
☐	Memory protection schemes cannot be implemented using page tables.
☐	Inverted page tables can be used to reduce the amount of memory required for page table.
☐	A translation look aside buffer (TLB) is implemented in hardware.

UNIVERSITY OF SKÖVDE

Name: ______________________________ Person number: ___________________

6.3 Which of the following statements about segmentation are true? (1 point)

☐	There is almost always external fragmentation when using segmentation to manage memory
☐	A segment may be used for the heap (dynamically allocated program memory).
☐	A virtual address consists of the tuple <segment_number, offset>.
☐	Data in a segment must belong to one process only.
☐	Memory management using segmentation is a dynamic storage allocation problem.

6.4 Which of following statements about contiguous memory allocation are true? (1 point)

☐	The operating system maintains information about allocated partitions and free partitions (holes).
☐	The first fit algorithm always assigns processes to the smallest available memory space.
☐	The multi-fit algorithm provides the optimal solution to contiguous memory allocation.
☐	External fragmentation wastes memory by creating free holes too small to be used by other processes.
☐	Contiguous memory allocation is a static storage allocation problem.

UNIVERSITY
OF SKÖVDE

Name: ______________________________ Person number: ____________________

6.5 Explain why memory pages (commonly 4KB) are much smaller than programs and how that helps memory management (3 Points)

UNIVERSITY
OF SKÖVDE

Name: ______________________________ Person number: __________________

6.6 Explain the consequences (both good and bad) of implementing hierarchical page tables. (3 Points)

UNIVERSITY
OF SKÖVDE

Name: ____________________________ Person number: __________________

7 Virtual memory (10 points)

7.1 Which of the following statements about virtual memory management are true? (1 point)

☐	A valid-invalid bit can be used in a page table to indicate if a page is memory resident.
☐	Pure demand paging is an inefficient use of system memory.
☐	Code needs to be in memory to execute, but an entire program does not.
☐	A page fault occurs when the wrong page is found in memory.
☐	Page fault service time is orders of magnitude greater than memory access time.

7.2 Which of the following statements about frame allocation are correct? (1 point)

☐	A high page fault rate can cause the operating system to spend more time replacing pages in memory than doing computation.
☐	An operating system does not need strategy for frame allocation.
☐	Equal allocation of frames increases the number of page faults for large programs.
☐	Local allocation of frames leads to inconsistent performance of individual processes.
☐	In a proportional allocation scheme the number of frames allocated to a process changes in relation to the number of processes loaded in memory.

UNIVERSITY
OF SKÖVDE

Name: ______________________ Person number: ______________

7.3 Which of the following statements about page replacement algorithms are true? (1 point)

☐	Page buffering can be used to cache pages so that recently used pages can be reloaded more quickly.
☐	The most frequently used (MFU) algorithm considers pages with least use were most recently loaded into memory and are likely to be used soon.
☐	A victim frame is selected by a page replacement algorithm to be removed from physical memory.
☐	Belady's Anomaly is observed for some reference strings when the number of frames allocated to a process is increased when using the optimal (OPT) page replacement algorithm.
☐	The process unionisation algorithm implements a fair selection policy for victim frames..

7.4 Copy on write (CoW) is used to improve the speed and efficiency of forking processes. Explain how CoW works when a child process is created, and what happens when either the parent or child process modifies data (3 points)

UNIVERSITY
OF SKÖVDE

Name: ______________________ Person number: ______________

7.5 Given the page reference string: 3, 6, 6, 3, 1, 4, 3, 3, 2, 6, 1, 1, 5, 2, 7, 0, 4, 2, 7, 2

Assuming demand paging with a page table of three frames, how many page faults would occur with the following replacement algorithms?

a. FIFO replacement (2 points)
b. LRU replacement (2 points)

For each algorithm, write a page table showing the state of the page table following each request in the reference string. Clearly mark each request that causes a page fault, and state the total number of page faults for each algorithm.

UNIVERSITY
OF SKÖVDE

Name: ______________________________ Person number: ____________________

8 File systems (10 Points)

8.1 Which of the following statements about files and directories are true? (1 point)

☐	A file owned by the root user and root group with the Unix file permission 644 (in octal) can not be read by any other system user.
☐	An operating system allows multiple processes to open the same file simultaneously and each process maintains a pointer to the last read or write locate each process used within the file.
☐	File extensions, e.g. .exe, .txt, .c and .tex, are used by the operating system as hints about the file contents.
☐	An operating system needs a policy for deleting shared files.
☐	In a tree file structure two files in the same directory (folder) can have the same name.

8.2 Which of the following statements are true about disk scheduling? (1 point)

☐	First come first served (FCFS) is an inefficient disk scheduling algorithm.
☐	The shortest seek time first (SSTF) algorithm can lead to starvation for some processes.
☐	Disk scheduling algorithms are constrained by the physical properties of hard disk drives.
☐	The Scan disk scheduling algorithm services read and write data requests only when the head moves in one direction.
☐	The Look algorithms reduce the number of sectors the head visits compared to the Scan family of algorithms.

UNIVERSITY
OF SKÖVDE

Name: ______________________________ Person number: __________________

8.3 Which of the following statements about disk drives are true? (1 point)

☐	A disk drive can only read from one cylinder at a time.
☐	A physical disk can not be partitioned into volumes that are treated as logical disks.
☐	Rotational latency refers to the time taken for the read/write head to move between cylinders on a disk drive.
☐	RAID is used to reduce the consequences of disk failure.
☐	Disk drives store data in sectors that include an error correction code and header information.

8.4 Given a disk with 200 cylinders and the request queue (where numbers represent cylinders): 25, 186, 3, 21, 55, 114, 9, 130, 49, 162, 89.

If the head starts at cylinder 48, calculate the total number of cylinders the head moves through to service the queue using each of the following algorithms:

(a) SCAN (2 points)
(b) C-LOOK (2 points)

UNIVERSITY
OF SKÖVDE

Name: ____________________________ Person number: __________________

8.5 Some RAID systems are implemented using striping. Explain what striping is (1) and identify one advantage of striping (1) and one disadvantage (1).

UNIVERSITY
OF SKÖVDE

Name: ____________________________ Person number: __________________

9 Protection and security (10 Points)

9.1 Information security concerns threats to the confidentiality, integrity, and availability of data. Which of the following statements are true? (1 point)

☐	Humans are the weakest part of any security system.
☐	Authorisation is used to support the integrity of data.
☐	Role based access control is used to implement the principle of most privilege.
☐	Unix file permissions can help support the protection of confidentiality and integrity of data.
☐	A buffer overflow is an error on a disk that can be exploited to execute malicious code.

9.2 Which of the following statements about network security are true? (1 point)

☐	Cryptography is an important component of network security.
☐	In a man-in-the-middle attack the attacker masquerades as both participants in a conversation.
☐	A firewall can inspect and discard network packets.
☐	Port scanning can not be used to identify network services running on a computer.
☐	In a replay attack a valid message is delayed by the attacker.

UNIVERSITY
OF SKÖVDE

Name: ____________________________ Person number: _________________

9.3 Which of the following statements about cryptography are correct? (1 Point)

☐	Encrypted data is confidential if and only if authorised individuals have access to the decryption key.
☐	When using a digital signature, the sender uses their private key to encrypt a message and the receiver uses a public key to decrypt the message and confirm the sender's identity.
☐	Determining the prime factors of very large numbers is computationally trivial.
☐	Public key cryptography can be used to support the distribution of keys for a symmetric encryption algorithm.
☐	A digital certificate is a copy of a public key digitally signed by a trusted party such as a certification authority.

9.4 Draw an access matrix for four domains and three objects where two domains can read and write two objects, another domain can read all three objects, and the fourth domain is unable to access either object, but can use a printer. (3)

UNIVERSITY
OF SKÖVDE

Name: ____________________________ Person number: __________________

9.5 Explain what single factor authentication (SFA) is (2), and how two-factor and other multi-factor authentication (MFA) schemes improve on SFA (2).