OPERATING SYSTEMS I

First and Last Name: **……**

Year of Study: **……**

28th of January, 2023

No points granted by default; minimum points required to pass exam: ***50p***

EXERCISE 1: PROCESS DEADLOCK PREVENTION (20p)

1. In a system with a single type of resources, there are 8 processes with the following maximal requirements:

Process	P1	P2	P3	P4	P5	P6	P7	P8
MAX	17	19	21	9	13	15	18	12

a) Specify the minimal value for the total number of available resources, so that the state of the system is considered safe, in the following scenarios:

a1) an initial allocation of resources: `(5, 12, 17, 7, 5, 13, 16, 10)` (7.5p)
a2) an additional allocation of resources: `(10, 0, 0, 1, 0, 1, 1, 1)` (5p)

b) Considering an initial allocation of resources of `(5, 12, 17, 7, 5, 13, 16, 10)` and a total number of resources equal to the one determined at a1) earlier, answer the following:

b1) give an example of a request which would trigger a deadlock in the system (2.5p)
b2) two processes request an additional 4 resources, with the state of the system still being safe. One of these processes is **P7**. Find the other process and the maximum number of resources it accessed. (5p)

EXERCISE 2: PROCESS SCHEDULING (35p)

2. A system with 10 processes needs to be scheduled according to the algorithms below. Using the information below, fill in the missing parts for all algorithms presented and answer the questions below.

			RR		SJF		SRTN		FCFS	
	Arrival	**Burst**	T_W	T_R	T_W	T_R	T_W	T_R	T_W	T_R
P1	9	11								
P2	15	19								
P3	12	13								
P4	13	31								
P5	30	27								
P6	0	5								
P7	3	12								
P8	3	7								
P9	0	3								
P10	3	8								

a) Find out the waiting time and the turnaround time for each of the processes through the GANTT diagram, for each of the algorithms presented. Wherever necessary, the quantum value (q) is the optimal one. (20p)
b) Which algorithm offers the best performance, taking into account that q is optimal, based on the average values for the waiting time and running time? Explain briefly the performance gains. (5p)
c) Which algorithm offers the lowest number of context switches in the situation above, considering that q is optimal? (2.5p)
d) What would happen if q = 25 in terms of response times and context switches, for the Round-Robin algorithm? Explain briefly which situation is more favorable and why. (2.5p)
e) Write down the process queue when scheduling occurs at the moment of time 30, in the SRTN algorithm. (2.5p)
f) In the SJF algorithm, besides the first process to execute, which process has the smallest response time? (2.5p)

EXERCISE 3: DEADLOCKS (40p)

3. In a system with 4 types of resources (printer, scanner, USB port, audio output), there are 5 processes with the following specifics:

Existing resources: **E = ?**

Available resources: **A = ?**

Current allocation:

$$\mathbf{C} = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 2 & 0 & 1 & 1 \\ 2 & 1 & 2 & 0 \\ 0 & 0 & 1 & 1 \\ 2 & 2 & 0 & 1 \end{pmatrix}$$

Requested:

$$\mathbf{R} = \begin{pmatrix} 3 & 1 & 2 & 1 \\ 2 & 2 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 2 & 0 & 0 & 2 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

a) Determine the total number of available resources **A** in the system if **E** = `(11, 9, 7, 8)`. (2.5p)
b) Determine the minimum number of resources required, for each resource type, so that the system state is still considered safe, if processes 1 and 2 make the following requests: `(1 1 1 1)` and, respectively, `(0, 2, 0, 1)` (15p)
c) Given **E** being the same as at point a) above, give an example of a request that would trigger a deadlock. (2.5p)
d) With **E** being the same as at point a) above, and an initial request for process 1 being `(3 1 1 1)`, as well as an initial allocation for process 4 of `(1 0 0 2)`, find all possible additional requests moving forward so that the state of the system is still considered safe. (20p)

EXERCISE 4: MEMORY MANAGEMENT (20p)

4. A system using a 16-bit Von Neumann architecture has a page size of 4,096 bytes and 16 KB of RAM memory. Access to the pages of the system is happening in the following order: 4, 4, 2, 1, 6, 7, 0, 1, 3, 4, 1, 2, 2, 1, 0, 7, 3, 3, 1, 5, 6, 6, 5, 5, 5, 1, 5, 1, 6.

a) What is the total number of virtual pages and page frames in this system? (1.25p)
b) How many page faults are issued when using the clock, FIFO and optimal algorithms for the accesses above? (15p)
c) Describe the memory mapping of the pages at moment 15 in the scenario above for all three algorithms mentioned. (1.25p)
d) Considering that at moment 15 above you are accessing the virtual address 0xDEAD, what is its corresponding page frame? (2.5p)

EXERCISE 5: CHALLENGES & QUESTIONS (35p)

5. Indicate whether each of the statements below is true or false, and explain your answer *briefly* (10p):

a) Non-preemptive scheduling algorithms cannot schedule 3 processes on a CPU with one single core. (1.5p)
b) A process can voluntarily bypass pre-emption in interactive operating systems, to allow other processes to run. (1.5p)
c) Threads in a process could be scheduled by the operating system independently of processes themselves. (1.5p)
d) Page faults are the reason why an interactive operating system is slowing down permanently over time. (1.5p)
e) Page faults happen more often on mobile operating systems with a lot of physical memory. (1.5p)
f) On multi-processor systems, it's required to protect data reads to memory with critical regions to avoid data corruption. (2.5p)

Answer the following questions *briefly* (25p):
g) Suppose a friend told you: "When designing a browser like Chrome, it's better to use a single thread per multiple tabs open, rather than a single process per tab". Explain if your friend is right, wrong or both. (5p)
h) The following code snippets modify two variables ***N*** and ***M*** (which are initialized with the values N=3, M=5) in 3 different threads, on a system with 1 CPU, 4 cores and 8 threads:

Thread 0	*Thread 1*	*Thread 2*
`M += N/3;`	`M -= N+2;`	`N /= M*2;`

h.1) Compute the expected result for the variables ***N*** and ***M*** after all threads finish execution in an interactive system, assuming each of the assignment instructions above are atomic instructions. (7.5p)
h.2) Assuming that thread 0 always executes last, and thread 1 has a higher priority than thread 2, compute the value of ***N*** and ***M*** after the execution of the threads complete in an interactive system. (2.5p)

i) Suppose a friend told you: "An operating system designed for autonomous driving (e.g. as in Tesla cars) allocates CPU-bound processes to a higher priority than I/O-bound processes." Explain whether your friend is correct. (5p)
j) Give an example of a situation when a low-priority process could prevent the execution of a higher-priority process. (2.5p)
k) If a process has two threads, one with the lowest priority, and the other with the highest priority, what is the priority of the process itself? (2.5p)