Name	Signature	Student Id Nr	*Points*

Operating Systems, miniexam 2, 13.2.2019 (12p)

Write your answer on this exam paper in the space given. Please notice, that the <u>exam paper is 2-sided</u>.

a) [3 p] Assume that thread P uses critical sections A and B every now and then. Sometimes it uses only A, and sometimes only B. Sometimes while using A it also needs B for a while. Sometimes while using B it also needs A for a while. Thread Q works the same way. Give a locking scenario for P and Q.

How can you prevent deadlock completely for P and Q?

Do you need to modify the code for both threads? Explain.

How can you prove that your solution works?

b) [1 p, do this last] Assume that two threads P and Q use the same critical section CS every now and then. Sometimes P suspends to wait for message from Q. Give a locking scenario for P and Q.

How can you prevent deadlock completely for P and Q?

c) [4 p] Clock algorithm selects one frame for replacement for a newly referenced page.
Why is CLOCK better than FIFO (First In - First Out)?

What data does CLOCK base its decision on? Who updates that data and when?

When will CLOCK start and when does it terminate?

What problem is there with CLOCK and how should CLOCK be modified to alleviate that problem?

d) [4 p] PFF (Page Fault Frequency) algorithm adjusts dynamically the number of page frames allocated to a process. When is PFF run?

What data does PFF base its decisions on? Who updates that data and when?

When and how will PFF adjust the number of frames for process?

Why you cannot use PFF with CLOCK?