

Name	Signature	Student Id Nr	Points

## Operating Systems, miniexam 2, 15.2.2016 (6p)

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

- a) [2 p] An application is implemented in 3 threads. Thread A executes the main calculation work. Thread B displays current application status continuously to the operator. Thread C save intermediate results continuously to the disk. The system has one processor (core). Would it be better, the threads are implemented as kernel level threads (KLT) or as user level threads (ULT)? Why?

Assume now that system has 4 cores. Would it now be better, the threads are implemented as KLT or ULT? Why?

- b) [1 p] The initial value of shared variable Sum is zero (0). Four threads A, B, C and D will each execute eventually (machine language) code

```

100:  ...
101:  LOAD R1, Sum      ; R1 ← mem(Sum)
102:  ADD R1, =1       ; R1++
103:  STORE R1, Sum    ; mem(sum) ← Ri
104:  ...

```

The intent is that each thread will increment Sum by one, and that its final value is 4. However, the program does not work properly in all scenarios. Give a scenario where final value for Sum is 2.

c) [1 p] What is wrong with the code in part (b)? How should you modify it, so that it would work correctly in all scenarios?

d) [2 p] Semaphores. Process P is implemented with 4 threads: A, B1, B2, and B3. Threads Bi will run their initialization code and then wait in (code position) Binit, until thread A has completed its own initialization and reached Ainit. Thread A will then wait in Ainit until all threads Bi have completed their calculations in Bdone. After this all threads will run into completion.

Give the solution (as pseudocode for threads A, B1, B2, and B3) for this synchronization problem with semaphores. Remember to give initial values for your semaphores.