

# BLG546E - Object Oriented Concurrent Programming

---

29.05.2013 - FINAL

Duration: 120 min.

- 1) (20p) Answer the following questions about consensus
  - a. What is a consensus number?
  - b. Give the consensus numbers of the following Java `AtomicReference` methods: `get()`, `getAndSet()`, `compareAndSet()`
  - c. Explain univalency and bi-valency concepts.
- 2) (20p) Answer the following questions about Linked List parallelization
  - a. Explain optimistic locking by comparing it to hand-over hand locking and lock-free parallelization methods. When we shall prefer optimistic locking?
  - b. Why do we need validation for optimistic locking? Give an example where validation is needed and explain.
  - c. Explain why do we need to use `AtomicMarkableReference()` in lock-free linked-list parallelization.
- 3) (40p) Illustrate the execution of a tournament tree barrier by redrawing the barrier for each finishing thread. Winner-loser conditions of threads are determined by priority numbers(smaller number indicates higher priority) provided below. The order of the thread completions are also provided below. Threads are assigned to tree leaves from left to right keeping the order of their names(A is assigned to leftmost leaf and J to rightmost).

Thread priorities:

A	B	C	D	E	F	G	H	I	J
0	9	4	6	5	2	8	1	7	3

Thread completion order:

**B - H - C - F - J - G - D - E - I - A**

- 4) (20p) Explain the following lock insufficiencies by giving examples:
  - a. Composability
  - b. Lost-wakeup problem, also state the disadvantage of solving this problem with using `notifyAll()`