

# CAP - Exercises: register allocation (chapter 7)

Laure Gonnord

Nov. 2016

## EXERCISE ► **Code production and register allocation**

Consider the expression  $E = ((n * (n + 1)) + (2 * n))$ . We assume that we have :

- A multiplication instruction `mul t1,t2,t3` that computes  $t1 := t2 * t3$ .
- A “immediate load” instruction `ldi t1 4`.
- The variable  $n$  is stored in the stack slot referred as  $[n]$  in the load instruction.

1. Generate a 3 address-code with temporaries and `ldr` instruction to load  $n$ . Do it as blindly as possible (no temporary recycling).
2. (Without applying liveness analysis) Draw the liveness intervals. How many registers are sufficient to compute this expression ?
3. Draw the interference graph (nodes are variables, edges are liveness conflicts).
4. Color this graph with three colors using the algorithm seen in the course ([http://laure.gonnord.org/pro/teaching/MIF08\\_Compil1617/07-RegisterAlloc.pdf](http://laure.gonnord.org/pro/teaching/MIF08_Compil1617/07-RegisterAlloc.pdf), slides 27-30).
5. Give a register allocation with  $K = 2$  registers using the *iterative* register allocation algorithm seen in course.