CS4212 – Compiler Design
School of Computing
National University of Singapore

September 1, 2012

Recitation 2

(handout used in class on September 6, 2012)

The instructor will guide the class towards developing solutions to the exercises given below. The students are strongly encouraged to attempt the exercises on their own before coming to class, and be ready to discuss/challenge the approach suggested by the instructor, as well as contribute their own ideas.

Exercise 1

The expression compiler introduced in Lecture 3 does not take advantage of the 6 general purpose registers. A way to correct that would be to imagine a virtual stack whose top 5-6 elements are held in registers (due to the peculiar behavior of the `idivl` instruction, you we need to keep register `%eax` available to hold temporary data, and it may not be feasible to make it part of the stack), and the rest of the virtual stack should be held in the actual stack. In this way, the most recent operands are always held in registers; this may also avoid redundant pushes and pops.

Exercise 2

After completing Exercise 1, generate code for several programs, and try to discover redundant code fragments. Then, implement a procedure that eliminates these fragments.