MC97 Homework 3

Due April 24, 1996; Max Hailperin

Do Exercise 10.1 on pages 711–712 of the textbook. Assume that `a`, `b`, and `c` are indexed from 1 to `n` in each dimension. You may use MIPS assembly language notation in place of the book's three-address intermediate code notation, if you are more comfortable with that. You should assume (in this case) that scalar variables live in registers. You can assume the existence of labels `a`, `b`, and `c` at the base of each array, and use the notation `(a-4)` to indicate the (constant) address four bytes before the start of `a`, etc. Even if you use the book's three-address intermediate code format, I would prefer that you not use the bracket notation, but rather did the address addition explicitly. If you use MIPS assembly language notation in place of the three-address intermediate code, you can skip parts b and h, which call for generating "target-machine code." If, on the other hand, you use three-address intermediate code, then go ahead and do parts b and h but using the MIPS as your target machine, rather than the one defined in the book.

Also, briefly describe any major opportunities for optimizing this routine that are not covered by the exercise.