Review Set 6

This Review Set asks you to prepare written answers to questions on local and global optimizations. Each of the questions has a short answer. You may discuss this Review Set with other students and work on the problems together.

1. Consider the rules for the constant propagation algorithm discussed in class:

   $$(1) \quad C_{in}(x, s) = \# \quad \Rightarrow \quad C_{out}(x, s) = \#$$
   $$(2) \quad C_{in}(x, x \leftarrow c) \quad \Rightarrow \quad C_{out}(x, x \leftarrow c) = c \quad (c \text{ is a constant})$$
   $$(3) \quad C_{out}(x, x \leftarrow f(\ldots)) = *$$
   $$(4) \quad E(x) \neq E(y) \quad \Rightarrow \quad C_{out}(x, y \leftarrow \ldots) = C_{in}(x, y \leftarrow \ldots)$$
   $$(5) \quad C_{in}(x, s) = lub\{C_{out}(x, p) \mid p \text{ is a predecessor of } s\}$$

   Note that these rules are not necessarily applied in order—the numbers are just there for ease of referencing. See the "Dataflow Analysis, Global Optimizations" lecture notes for the actual steps of the algorithm.

   (a) Give a concise English description for each of rules 1–4.

   (b) Note that rules 1–4 define C_{out} in terms of C_{in}. Rule 5, on the other hand, defines C_{in} based on the C_{out} values of all predecessor statements. Give two distinct examples that show some set of predecessor C_{out}'s and the resulting C_{in} computation.

   (c) Briefly explain why the algorithm (as described in the lecture notes) is guaranteed to terminate.

   (d) Note that, in rule 4, we set $y \leftarrow \ldots$ (where "$\ldots$" is some expression e). Why is it safe to assume that the evaluation of e does not change the value of x?

2. Consider the following fragment of intermediate code:

```
      START
      if a = 2 goto L3
L0:  b := 2
L1:  d := a / 2
      c := a % b
      if c = 0 goto L2
      if b >= d goto L3
      b := b + 1
      goto L1
L2:  a := a + 1
      goto L0
L3:  END
```

(a) Divide this code into basic blocks; there should be at least 6. Assume that START and END are placeholder instructions (i.e. they don't do anything).

(b) Draw a control-flow graph for this program, using your answer to (a). Place each basic block in a single node.

(c) Annotate your control-flow graph with the set of variables that are live before and after each statement. **Assume that only a is live at the entry to L3**.

(d) Describe concisely what this program does if the value of a is the only output.