

Homework 5CS 540
Spring 1991
Craig A. Rich

1 Consider the following fragment of a Pascal program:

```
i := 1;  
repeat  
  x := a[i];  
  y := a[i + 1];  
  i := i + 1  
until x < y or i > 100
```

- a) Translate the fragment into a three-address program and construct the flow graph which corresponds to it. Assume the array *a* contains 4-byte elements starting at offset 0, *i* does not occur elsewhere, and short-circuit boolean evaluation is not used. Every statement except the first should contain one arithmetic, boolean, relational, or indexing operation; therefore, 5 temporary variables should be introduced.
- b) Apply strength reduction to the flow graph loop in a). Show all induction variables and the resulting flow graph.
- c) Apply induction variable elimination to the flow graph loop in b). Show the resulting flow graph.
- d) Apply the DAG basic block optimization to each block of the flow graph in c). You need not show the DAG's constructed, but show the resulting flow graph.

2 Consider the flow graph constructed in Homework 1, problem 2a).

- a) Apply algorithmic reaching definition analysis to the flow graph, showing the values of IN[B] and OUT[B] after each while-loop iteration. Process blocks in their physical order (which is a depth-first order).
- b) Apply algorithmic live variable analysis to the flow graph, showing the values of OUT[B] and IN[B] after each while-loop iteration. Process blocks in reverse physical order.

3 Let (D, \leq) be a lattice, $x, y, z \in D$. Prove the following lemmas:

a) **Lemma.** $x \wedge y \leq x$ and $x \wedge y \leq y$.

b) **Lemma.** $x \leq y$ and $x \leq z \iff x \leq y \wedge z$.

c) **Lemma.** $x \leq y \iff x = x \wedge y$.

d) **Lemma.** $x \leq y \wedge z \implies x \leq z$.

e) **Lemma.** $x \leq y \implies x \wedge z \leq y \wedge z$.