Computer Science 320: Final Examination

May 17, 2001

You have 3 hours to answer the following questions. This midterm is closed book/closed notes. For partial credit, show all work. State all assumptions. All work must be done in your exam booklet. Write out and sign the Honor Code pledge before turning in the test.

"I pledge my honor that I have not violated the Honor Code during this examination."

Problem A - Lexical Analysis

Do the following:

1. Write a regular expression that defines a fixed-decimal number with no leading or trailing zeros. That is, 0.0, 123.01, and 1230005.0 are legal, but 00.0, 001.000, and 0002345.1000 are illegal.

2. Build a *Deterministic Finite Automaton* (DFA) that recognizes this regular expression.

3. In five (5) sentences or less, discuss why the lexical analysis phase of a compiler is implemented separately, using a DFA.

Problem B - Grammars

1. Is the following grammar ambiguous? Prove your answer.

$$S \rightarrow \; (\; S \;)\; S \qquad\qquad S \rightarrow \;)\; S \;(\; S$$
$$S \rightarrow \; \text{X} \qquad\qquad S \rightarrow \epsilon$$

2. Construct the SLR parsing table for the following grammar.

$$E \rightarrow E + T \qquad E \rightarrow T$$
$$T \rightarrow T F \qquad T \rightarrow F$$
$$F \rightarrow F * \qquad F \rightarrow \; \text{a} \qquad F \rightarrow \; \text{b}$$

3. You've joined a start-up named "LawnBeGone, Inc." which has just won venture capital to make an automatic lawn mower. The unit is to operate by tracing a predefined path around the customer's yard. As the only computer scientist on the team, it is your responsibility to make it programmable.

   The programming language has only two commands. Here is a sample program:

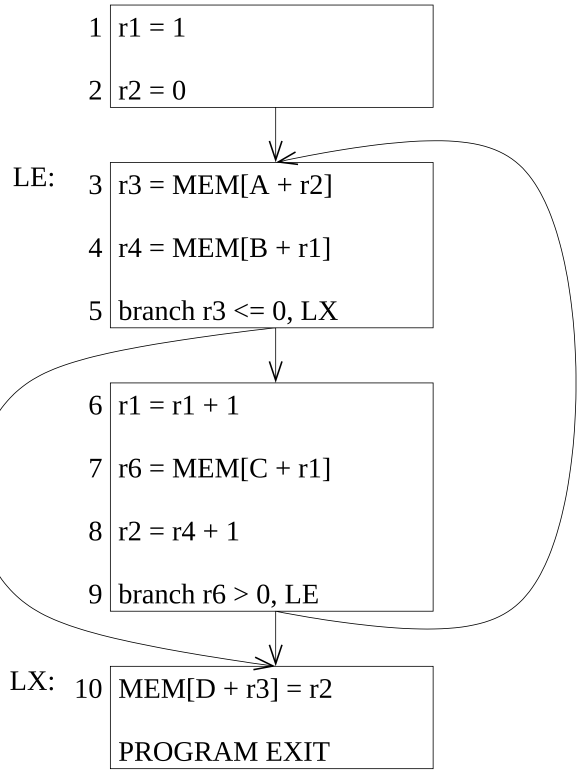
```
Forward(100);
Rotate(90);
Forward(1);
Rotate(90);
Forward(100);
```

   Without using lex or yacc, write the code (ML, C, or psuedo-code) to interpret programs in this language for the lawn mower. The machinery is directed by calls to the `M_forward(feet)`, `M_rotate(degrees)`, and `M_powerOff()` functions. The code can be read through the `P_getChars(num)` function which returns the next `num` non-white-space characters. If an error in the program is encountered, power the unit off.

Problem C - Garbage Collection

1. What is garbage collection? (2 sentences or less)
2. Describe the mark and sweep technique. (4 sentences or less)
3. You will have to fly in what you build. Would you use a garbage collecting language for your rocket's stability control system? Why or why not?

Problem D - Live Variable Analysis/Register Allocation



Do the following:

1. Compute the LIVE IN and LIVE OUT sets for instructions 1 $\rightarrow$ 10 in the following code segment. Show each step of the dataflow analysis.
2. Show the register interference graph.
3. Register allocate the code with the fewest number of registers.
4. Suggest a change in the insruction order which would reduce the number of necessary registers.

Problem E - Optimization

You're the professor for a compiler construction class, and you have to write a final exam question on code optimization. Create a small code example for the exam which reduces to:

```
return 100
```

after the following optimizations are applied in this order:

1. Dead Code Elimination
2. Common Subexpression Elimination
3. Copy Propagation
4. Constant Propagation
5. Constant Folding
6. Constant Propagation

Create the answer key by showing the code before and after each optimization step. Remember, the smaller the original code example, the easier it is for you to grade it.

Problem F - Scheduling

You are a compiler designer working to target a new processor. Unfortunately(?), you haven't signed the company's NDA (non-disclosure agreement). To enable you to write the scheduler for the new machine, the company decides to give you a list of instructions and the resources each instruction consumes. However, they won't tell you what the resources are physically. Instead, they call the resources X, Y, and Z. They will tell you that all resources are fully pipelined.

Instruction Characteristics:

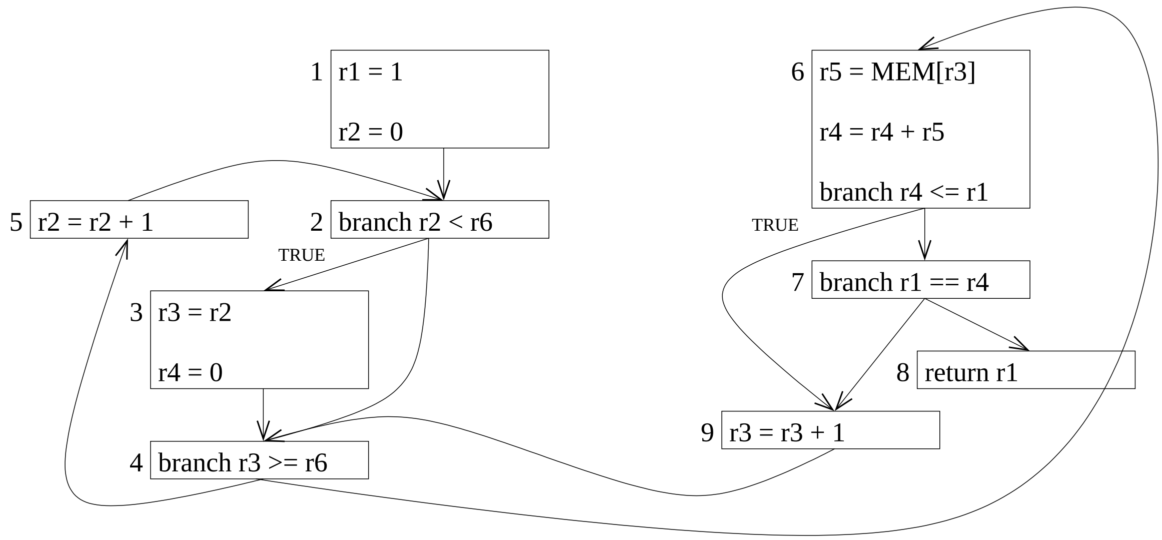
Instruction	Latency	Resource Consumption
Add/Sub	1 cycle	1 X, 2 Y
Multiply	2 cycles	1 X, 3 Y
Divide	3 cycles	2 X, 1 Y
Load	2 cycles	1 X, 2 Z

For the code segment:

1. Draw the dependence graph. Clearly label each edge with its latency and type.
2. What additional information do you need to schedule for their machine?
3. Show the top-down list schedule of the code for a machine of infinite resources.
4. What are the minimum machine resources necessary to complete the code in the fastest possible time?

```
r1 = MEM[A]
r2 = MEM[B]
r3 = MEM[C]
r4 = MEM[D]
r5 = r1 / r2
r6 = r3 * r4
r7 = r5 + r6
r8 = r1 + 333
r9 = r2 - r8
```

Problem G - Static Single Assignment



Convert this program to SSA form. Show your work after each stage.

1. Add a start node containing initializations of all registers.
2. Draw the dominator tree.
3. Calculate dominance frontiers.
4. Insert ϕ-functions and add subscripts to registers.