T-106.550 Compilers
Exam Dec. 18, 2006

Examiner: Jorma Tarhio

No written material is allowed in this exam. Submit at least one answer sheet, even if an empty one! Write on *each* answer sheet you submit the code of the course, the date, your name, and your student ID number.

You are given an extra point if you fill the (anonymous) course evaluation on the homepage of the course not later than on Dec. 31.

---

1. Answer shortly to the following questions:

   (a) What does the letter R in LR(1) mean?

   (b) What are L- and R-values?

   (c) What is dead code?

   (d) What is bottom-up parsing?

   (e) What does the k mean in LL(k) grammars?

   (f) What is constant folding?

   (6 p)

2. Consider the following grammar:

$$
\begin{array}{lcl}
L & \rightarrow & L\,;\,S \\
L & \rightarrow & S \\
S & \rightarrow & B \\
S & \rightarrow & \text{if } B \text{ then } S \text{ else } S \text{ fi} \\
S & \rightarrow & \text{if } B \text{ then } S \text{ fi} \\
B & \rightarrow & \text{id} = \text{num}
\end{array}
$$

   Transform the grammar to the LL(1) form. (6 p)

3. (a) Consider the following grammar:

$$B \rightarrow B \downarrow B \mid B \uparrow B \mid (B) \mid \textbf{id}$$

   Show that the grammar is ambiguous. (2 p)

   (b) Eliminate the ambiguity from the grammar by using the following rules:

   - Operators $\downarrow$ and $\uparrow$ are right associative.
   - The precedences are from the weakest to the strongest: $\downarrow$, $\uparrow$, and ().

   (2 p)

   (c) Give an example of unambiguous grammar that is not LR(1). (2 p)

P.T.O.

4. The costs and intermediate language patterns of machine instructions are:

```
ADD ri, rj, rk     1    BINOP(+, TEMP(rj), TEMP(rk))
ADDI ri, rj, c     2    BINOP(+, TEMP(rj), CONST(c))
ADDI ri, r0, c     1    CONST(c)
LOAD ri, rj[c]     2    MEM(BINOP(+, TEMP(rj), CONST(c)))
LOAD ri, rj[0]     2    MEM(TEMP(rj))
LOAD ri, c         4    MEM(CONST(c))
STORE rj[0], ri    3    MOVE(MEM(TEMP(rj)), TEMP(ri))
MOVEM rj, ri       2    MOVE(MEM(TEMP(rj)), MEM(TEMP(ri)))
```

By using dynamic programming, give the lowest-cost machine code for the following piece of intermediate code. Give all steps of your analysis.

```
MOVE(MEM(BINOP(+, MEM(CONST(a)), CONST(b))),
     MEM(TEMP(ra)))
```

(6 p)

5. Consider the following program:

```
    i = a + b
L1: c = a*b
    i = i - 1
L2: d = a/i
    if d = 10 goto L3
    if d > b goto L1
    c = c + 1
    if c > i goto L2
L3: print c
```

Divide the program to basic blocks. Draw the domination tree of the program. Use it to find out the natural loops and list the basic blocks of each natural loop. Which of the loops is the innermost one? (6 p)