CMSC 430 FINAL
December 14, 2001 8-10am

Do not turn the page until told to do so.

Answer all questions in exam book.

1 [10]. Show that the following grammar is LALR(1). Give the parsing table.

S → X1 | YZ
X → 0X | 2
Y → 0Y | 2
Z → Z3 | 3

2 [10]. Consider the regular expression: 0*01 | 10*1

(a) Give the deterministic finite state automaton that recognizes the same set.
(b) Give the regular grammar that accepts the same set.

For problems 3 to 7, assume the following machine architecture:

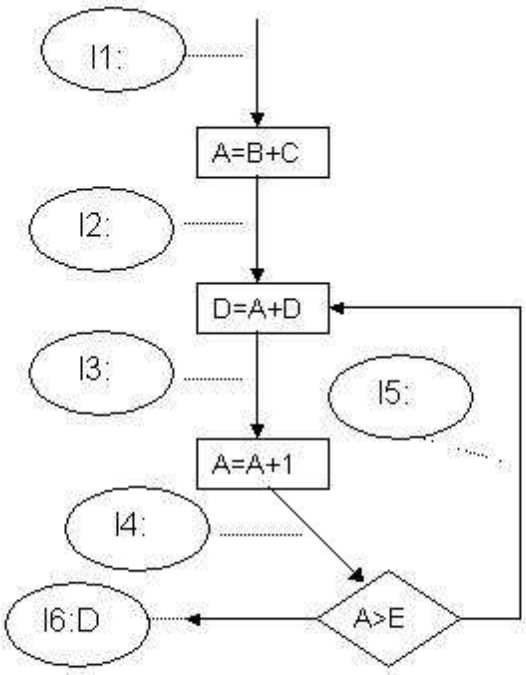
Instruction	Meaning	Exec time
load reg,memory	Load memory contents into register	4
store reg,memory	Store register contents at memory	4
add reg1,reg2,reg3	reg3 = reg1+reg2	1
sub reg1,reg2,reg3	reg3 = reg1-reg2	1
jpos reg,address	Jump to address if reg contents>0	3
jneg reg,address	Jump to address if reg contents < 0	3
jump address	Jump to address	2
Nop	Do nothing	1

- An instruction can start each cycle if the operands are available.
- Each instruction is 4 bytes long
- The machine has 4 registers: R0 .. R3.

3 [10]. Consider the flowchart following this question. (Note, if you know how to do question 4, you'll save time doing 4 first and then doing 3.)

(a) What does it mean that each oval represents variables that are live along that arc (i.e., what does liveness mean)?

(b) Fill in the values for ovals I1 through I5. The output liveness property for I6 is already given as the variable D.



4 [10]. Use the fixed point theorem for lattices to derive the liveness properties for question 3. Show all work. (Note, you'll save some time if you do this question and simply write in your answers for question 3.)

5[10]. Use register coloring to assign registers for each variable or temporary in the program of question 3. Show all work, including the interference graph.

6 [10]. Consider the statements:

A = B+C;
D = C+E;
E = A+D;
F = B+C;
G = C+E;

(a) Use value numbering to find common subexpressions in the above statements.(Show all work.)
(b) Give the machine language instructions that minimize execution time for the optimized set of statements (i.e., with common subexpressions removed).

7 [10]. Given the program sequence:

```
while A>B do B=B+1;
```

Give the sequence of machine language statements generated to execute this fragment in the order that a one-pass compiler will generate them (e.g., such as putting the code generator into JCUP or YACC). Assume the first instruction will be at location 1000. (Remember each instruction takes 4 bytes and the instructions are given before question 3.)

8 [10].

(a) Draw the DAG for the statement:

```
A = (B+C)*D+(B+C)
```

(b) Assume A is at location 4 in stack frame, B is at 8, C is at 16, and D is at 20. Give the Java bytecodes to execute this statement. (Opcodes can be given symbolically.) [Full credit if sequence is optimized for space – least number of bytes for code.]

9[10]. Each integer takes 4 bytes.

(a) Assume array A[1:10, 3:20] begins at location 1000. Give the accessing formula to access A[I,J].
(b) Define B[1:18] as the slice A[2,*] (i.e., B[K] = A[2,K+2].) Give the accessing formula for B[K].

10[10]. For stack frames:

(a) What is the static link?
(b) What is the dynamic link?
(c) Which is not needed in Java and why?
(d) Why does the inclusion of nested blocks (i.e., structures like {int x; x=a+b;}) not change your answer to (c)?
(e) What is a display equivalent to in the above list and why?