

Problem 1: Using the SPARC Architecture Manual (SAM) V8 answer the questions below. The SPARC Architecture Manual is distributed with the source for the microSPARC IIep in directory `.../models/sparc_v8/docs/pdf` of the distribution which can be downloaded from <http://www.sun.com/microelectronics/communitysource/sparcv8/>.

Alternate instructions will be given in class.

The SAM is 295 pages, so don't print it all out. It is not necessary that you understand everything in the SAM to answer these questions. See Appendix B to answer the last question.

- What size integers does the ISA support?
- What size floating-point numbers does the ISA support?
- How many floating-point registers does the ISA support, how large are they, and how are the different-sized FP numbers placed in them?
- What is the binary coding of the following SPARC v8 instruction:

```
ldsh [%r8 + 2], %r9    ! Load signed half, r9 = Mem[r8 + 2]
```

Problem 2: Find the static and dynamic instruction count for the DLX program below. (DLX is described in Chapter 2 of the text and summarized in the last two pages. Comments, preceded by a `!`, describe what the instructions do.) The program adds up a table of numbers.

```
lhi  r2, #0x1234      ! Load high: r2 = 0x12340000
ori  r2, r2, #0x5678 ! r2 = r2 0x5678
addi r4, r0, #10      ! r4 = r0 + 10,  r0 always = 0
sub  r3, r3, r3       ! r3 = 0.  There are lots of ways to do this!
LOOP:
lw   r1, 0(r2)        ! r1 = Mem[r2+0]
add  r3, r3, r1       ! r3 = r3 + r1
addi r2, r2, #4       ! r2 = r2 + 4
subi r4, r4, #1       ! r4 = r4 - 1
bneq r4, LOOP        ! if r4 != 0 goto LOOP
```

Problem 3: DLX does not allow arithmetic instructions to access memory. Suppose they could and suppose all the addressing modes in Figure 2.5 of the text were available. Re-write the program to use as few instructions as possible (but still perform the same function).

Problem 4: Find the static and dynamic instruction count of the program written for the question above.

Problem 5: What factors (relating to CPI and ϕ) would one have to take into account to compare the execution time using the dynamic instruction count of the original program and the re-written program?