

## Study Questions for Patterson & Hennesey, Section 2.5

### Section 2.5

This sheet emphasizes the material in the text that was not covered in lecture. Any possible quiz would cover the lecture material as well.

Figure 2.9 shows that there is one difference between C/C++ and Java: In Java for positive numbers, `>>` is a shift right, but for negative numbers, it does something which we will learn about later (called shift right arithmetic), and instead there is another operation, `>>>`, which shifts right logical on both positive and negative numbers. The Java `>>>` corresponds to the C and C++ operation `>>`, and corresponds to the MIPS `srl`. The text authors, coming from a C background, will elsewhere (e.g. on the green summary sheet) use `>>` when a Java programmer would use `>>>`.

1. Now we know what the shift field of an instruction is for. Write

```
sll $t0, $s0, 2
```

as a 32-bit word to show the shift field being used.

2. Is the `srl` instruction simply the `sll` instruction with a negative value for the shift amount? That is, are

```
srl $t0, $s0, 2
```

and

```
sll $t0, $s0, -2
```

the same instruction? [Hint: This is the first question where I ask you to reference the material in Appendix A.10. See pages A-55 to A-56.]

3. Why do you think that the designers of the MIPS machine did things the way that your answer to Question #2 says that they did?

4. I said that `sll` and `srl` are I-type instructions. What are the names of the corresponding R-type instructions? [See pages A-55 to A-56 again.]

5. What is a “mask.” [p.70]

6. `not` is a pseudo-op in MIPS. Using `nor`, write an instruction that does the same thing as you might imagine the following MIPS pseudo-op does:

```
not $t1, $t0 # $t1 = ! $t0
```