Friday questions week 4

1. Which are the five main steps in instruction execution on a RISC processor?

   page 147

2. In a five-stage pipeline of a processor which can issue (i.e. start executing) one instruction every clock cycle, what is the ideal performance improvement over non-pipelined instruction execution?

   page 148

3. During the 1970's most computers did not take advantage of pipelining. Why?

   page 147

4. Why can there be a pipeline stall after a load instruction but not after an add instruction?

   page 149

5. When more than one instruction starts executing every clock cycle, it becomes very likely that a conditional branch instruction will be executed in a clock cycle. It is important to start fetching new instructions to be executed *after* the branch but since the processor cannot know for sure whether a conditional branch will be taken or not, it must guess which instructions to fetch. What is the table used for this called?

   page 149

6. By guessing which instructions should be executed, a superscalar processor uses *speculative execution*. Consider a store instruction that is executing speculatively (i.e. before the processor knows for sure it should execute). Which mechanism is used to prevent the store from modifying memory before it certain it will be executed?

   page 150

7. What is the purpose of rename registers?

   page 150

8. What do the terms temporal and spatial locality mean?

   page 154