The following is a list of possible questions for our quiz on February 17th. Some of the questions will not be asked in the quiz. All the questions that will appear in the quiz will appear exactly as shown below (however, numeric parameters may be changed). The quiz is closed textbook, closed notes and closed neighbors. Note that the questions, which did not appear in this quiz, still may appear in the exams. You will find a solution for these questions during lectures.

#1: What are “structural hazards”? Show an example of the structural hazard (using assembly instructions).

#2: What are “data hazards”? Show an example of the data hazard (using assembly instructions).

#3: What are “control hazards”? Show an example of the control hazard (using assembly instructions).

#4: What are the four different types of data hazards?

#5: Show an example of RAR data hazards.

#6: Show an example of RAW data hazards.

#7: Show an example of WAR data hazards.

#8: Show an example of WAW data hazards.

#9: What are “static optimizations (i.e., “static instruction scheduling”)”?

#10: Who perform “static optimizations (i.e., “static instruction scheduling”)”?

#11: What are “dynamic optimizations (i.e., “dynamic instruction scheduling”)”?

#12: Who perform “dynamic optimizations (i.e., “dynamic instruction scheduling”)”?

#13: What is the counter concept of “run-time”?

#14: What type(s) of data hazards (WAR, RAW, and WAW) does “forwarding” will be effective (assume pipeline datapath processors)?

#15: Why is “forwarding” not effective for eliminating “control hazards”?

#16: Why is “forwarding” not effective for eliminating “control hazards”? 
#17: Is “forwarding” a dynamic or a static optimization technique (justify you answer)?

#18: Is “instruction scheduling by a compiler” a dynamic or a static optimization technique (justify you answer)?

#19: What type(s) of data hazards (WAR, RAW, and WAW) does “instruction scheduling by a compiler” will be effective (assume pipeline datapath processors)?

#20: Why is “instruction scheduling by a compiler” not effective for eliminating “control hazards”?

#21: What type(s) of pipeline hazards does “delayed branch” will be effective (assume pipeline datapath processors)?

#22: What is the major weakness in “delayed branch”?

#23: What are the major advantages in dynamic optimizations (i.e., “dynamic instruction scheduling”)?

#24: What are the major disadvantages in dynamic optimizations (i.e., “dynamic instruction scheduling”)?