P2. Dynamic scheduling. Consider two machines. (1) Dual issue scoreboard, with separate functional units for (a) integer ops, (b) loads and stores, (c) branches, (d) FP adds/subs, and (e) FP multiply/divide. The scoreboard machine can issue two instructions per cycle to functional units (if the units are available), and complete many in a cycle. (2) A single-issue Tomasulo architecture, which issues 1 instruction per cycle to a reservation station, has the same set of functional units as the scoreboard machine, but has 3 reservation stations per functional unit. Functional units are pipelined.

Assume for both machines latencies of a single cycle except for FP adds (6 cycles) and FP multiplies/divides (15 cycles). Assume both machines have full support for branch speculation (that is, they never have to stall waiting for branches to resolve), and that branches are predicted accurately. A latency of n cycles means that dependent instructions can be begin execution n cycles after the producer begins execution, and that the instruction is in the functional unit exactly n cycles.

For each loop below, indicate whether the scoreboard machine would be faster, Tomasulo would be faster, or they would be about the same (e.g., within roughly 20%).

L1:  
Lw  R1, 1000(R5)  
Add R6, R3, R1  
Lw  R4, 500(R6)  
Add R7, R1, R4  
Lw  R9, 2000(R7)  
Add R5, R9, R9  
Bne R5, R2, L1

(a)

L1:  
Lw  R1, 1000(R5)  
Add R6, R3, R8  
Lw  R4, 500(R6)  
Add R7, R8, R6  
Lw  R9, 2000(R8)  
Add R11, R4, R4  
Bne R4, R2, L1

(b)

L1:  
Ld  F4, 1000(R5)  
Mul.d F6, F2, F4  
Add R5, R5, R3  
Bne R5, R1, L1

(c)

L1:  
Ld  F4, 1000(R5)  
add.d F6, F2, F4  
add.d F8, F10, F4  
Add R5, R5, R3  
Bne R5, R1, L1

(d)

L1:  
Ld  F2, 1000(R5)  
Mul.d F6, F2, F6  
Add R5, R5, R3  
Bne R5, R1, L1

(e)