1. (50 points) Run the following code on Scoreboard with function units including two FP multipliers， one FP divider, one FP adder that can perform float add and sub operations, and two integer units that is responsible for memory accessing, integer ALU operations and branch).

(1) Fill the following tables with the scoreboard status at the end of **5th** clock cycle when the first L.D instruction just finish WB stage. The scoreboard status includes instruction status, function units status and FP register status.

```
L.D     F0,0(R1)      ; F0←MEM[R1+0]
L.D     F4,0(R2)      ; F4←MEM[R2+0]
MUL.D   F6,F0,F4      ; F6←F0 * F4
ADD.D   F8,F0,F2      ; F8←F0 + F2
S.D     F6, 0(R3)     ; MEM[R3 + 0] ← F6
S.D     F8, 0(R4)     ; MEM[R4 + 0] ← F8
```

Instruction producing result	Instruction consuming result	Latency (in clock cycle)
FP operation	FP operation	3
FP operation	FP store（S.D）	2
FP load（L.D）	FP opereation	1
FP load（L.D）	FP store（S.D）	0

Instructions	发射（Issue）clock	读操作数(RO) clock	执行（EXE）clock	写结果(WB) clock
L.D F0, 0(R1)	1	2	3-4	5
L.D F4, 0(R2)	2	3	4-5	
MUL.D F6,F0,F4	3			
ADD.D F8,F0,F2	4			
S.D F6, 0(R3)				
S.D F8, 0(R4)				

Function units	Function Unit Status									
	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	A
Integer1										
Integer2	Yes	Load	F4		R2				No	R2+0
FPMult1	Yes	Mul	F6	F0	F4		Int2	Yes	No	
FPMult2										
FPDivider										
FPAdder	Yes	Add	F8	F0	F2			Yes	Yes	

	Register Status								
	F0	F2	F4	F6	F8	F10	F12	.....	F30
FU	M[R1+0]		Int2	Mul1	FPAdder				

(2) Fill the following table with instruction status at the end of **6th** Clock Cycle as the first line in the table.

	发射（Issue）	读操作数(RO)	执行(EXE)	写结果(WB)
L.D F0, 0(R1)	1	2	3-4	5

L.D F4, 0(R2)	2	3	4-5	6
MUL.D F6,F0,F4	3			
ADD.D F8,F0,F2	4	6		
S.D F6, 0(R3)				
S.D F8, 0(R4)				

Function units	Function Unit Status									
	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	A
Integer1										
Integer2										
FPMult1	Yes	Mul	F6	F0	F4			Yes	Yes	
FPMult2										
FPDivider										
FPAdder	Yes	Add	F8	F0	F2			No	No	

	Register Status								
	F0	F2	F4	F6	F8	F10	F12	.....	F30
FU	M[R1+0]		M[R2+0]	Mul1	FPAdder				

2. （50 points）For the following instruction sequence:

L.D F6, 34(R2)
L.D F2, 45(R3)
MUL.D F0, F2, F4
SUB.D F8, F2, F6
DIV.D F10, F0, F6
ADD.D F6, F8, F2

(1) Fill in the tables that Tomasulo algorithm used when the first instruction completes and finishes writing result. Assume the memory access unit needs two clock cycles to do execution: one for address calculation and one for memory access.

Instruction status			
	ISSUE	EXECUTE	WRITE RESULT
1	1	2-3	4
2	2	3-4	
3	3		
4	4		
5			
6			

Reservation stations							
NAME	BUSY	Op	Vj	Vk	Qj	Qk	A
Load1							
Load2	Yes	Load		R3			R3+45
Add1	Yes	SUBD		M[R2+34]	Load2		
Add2							
Add3							
Mult1	Yes	MULTD		F4	Load2		
Mult2							

Register status									
Field	F0	F2	F4	F6	F8	F10	F12	…	F30
Qi	Mult1	Load2		M[R2+34]	Add1				

(2) Assume the following latencies: load is 1 clock cycle; add is 2 clock cycles; multiply is 10 clock cycles; divide is 40 clock cycles. Fill in the tables when the instruction MUL.D is about to write result (write result in next cycle).

Instruction status			
	ISSUE Clock	EXECUTE finish Clock	WRITE RESULT Clock
1	1	2	3
2	2	3	4
3	3	5-14	
4	4	5-6	7
5	5		
6	6	8-9	10

Reservation stations							
NAME	BUSY	Op	Vj	Vk	Qj	Qk	A
Load1							
Load2							
Add1							
Add2							
Add3							
Mult1	Yes	MULTD	M(R3[45])	F4			
Mult2	Yes	DIVD		M[R2+34]	Mult1		

Register status									
Field	F0	F2	F4	F6	F8	F10	F12	…	F30
Qi	Mult1	M[R3+45]		2M[R3+45]-M[R2+34]	M[R3+45]-M[R2+34]	Mult2			