CSE 560 – Practice Problem Set 6 Solution

1. Rename this instruction sequence:

mul r4, r5 → r1
add r1, r2 → r3

Map table	
r1	p1
r2	p2
r3	p3
r4	p4
r5	p5

Free-list
p6
p7
p8
p9
p10

mul p4, p5 → p6
add r1, r2 → r3

Map table	
r1	p6
r2	p2
r3	p3
r4	p4
r5	p5

mul p4, p5 → p6
add p6, p2 → p7

Map table	
r1	p6
r2	p2
r3	p7
r4	p4
r5	p5

Free-list
p8
p9
p10

2. Dispatch this instruction:

div p7, p6 → p1

Insn	Inp1	R	Inp2	R	Dst	Age

Ready bits	
p1	y
p2	y
p3	y
p4	y
p5	y
p6	n
p7	y
p8	y
p9	y

Insn	Inp1	R	Inp2	R	Dst	Age
div	p7	y	p6	n	p1	0

3. Determine which of the following instructions are ready.

Insn	Inp1	R	Inp2	R	Dst	Age
add	p3	y	p1	y	p2	0
mul	p2	n	p4	y	p5	1
div	p1	y	p5	n	p6	2
xor	p4	y	p1	y	p9	3

The add and xor instruction are ready to issue.

(a) Which will be issued on a 1-wide machine?

The add instruction will issue, because it has the smallest age (i.e., it is the oldest).

(b) Which will be issued on a 2-wide machine?

Both the add and the xor will issue on a 2-wide machine.

(c) What information will change if we issue the instruction from part (a)?

After the add instruction issues, the table will be revised to look as follows:

Insn	Inp1	R	Inp2	R	Dst	Age
mul	p2	y	p4	y	p5	1
div	p1	y	p5	n	p6	2
xor	p4	y	p1	y	p9	3

and the mul instruction is ready to issue (this assumes the add latency is just one clock).

4. Using the revised pipeline diagrams presented in class, show the execution of the following instruction sequence:

```
div   r2 ← r3, r5
add   r1 ← r2, r4
mul   r4 ← r6, r6
```

You should assume that the execution units for the three instructions are as follows:

4 clocks for an add
10 clocks for a multiply
20 clocks for a divide

Start by showing the instructions after renaming, and then show the pipeline diagram for a dual-issue processor.

Assuming the map table starts out as below:

Map table	
r1	p1
r2	p2
r3	p3
r4	p4
r5	p5
r6	p6

The renamed registers are as follows:

div p7 ← p3, p5
add p8 ← p7, p4
mul p9 ← p6, p6

and the pipeline diagram becomes:

Instruction		Disp	Issue	WB	Commit
div	p7 ← p3, p5	1	2	22	23
add	p8 ← p7, p4	1	22	26	27
mul	p9 ← p6, p6	2	3	13	27