CSE 320 Spring 2014

Name: SOLUTION ID:

Homework #6 Quiz Version B

No calculators, notes, or textbooks allowed. Show all your work for full credit.
Time limit: 15 mins

Consider the following MIPS code

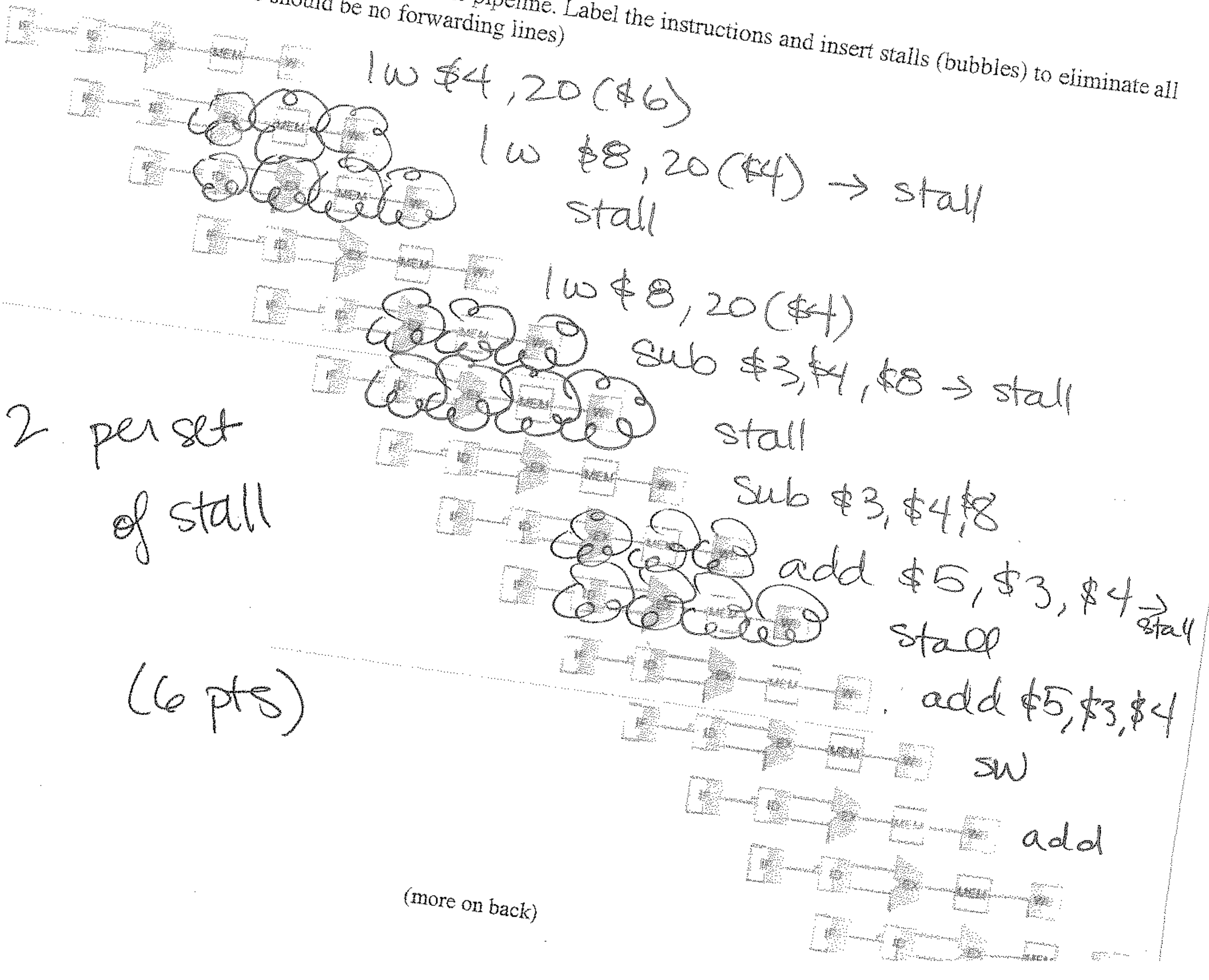
```
lw  $4, 20 ($6)
lw  $8, 20 ($4)
sub $3, $4, $8
add $5, $3, $4
sw  $3, 20 ($6)
add $3, $7, $8.
```

(a) What type of hazards occurs in the code above? Where do they occur?

+2 Data hazards

lw/lw $4
lw/sub $8
sub/lw $4
add|sub $3

(b) Assume, there is **no forwarding** in the pipeline. Label the instructions and insert stalls (bubbles) to eliminate all hazards (ie. There should be no forwarding lines)



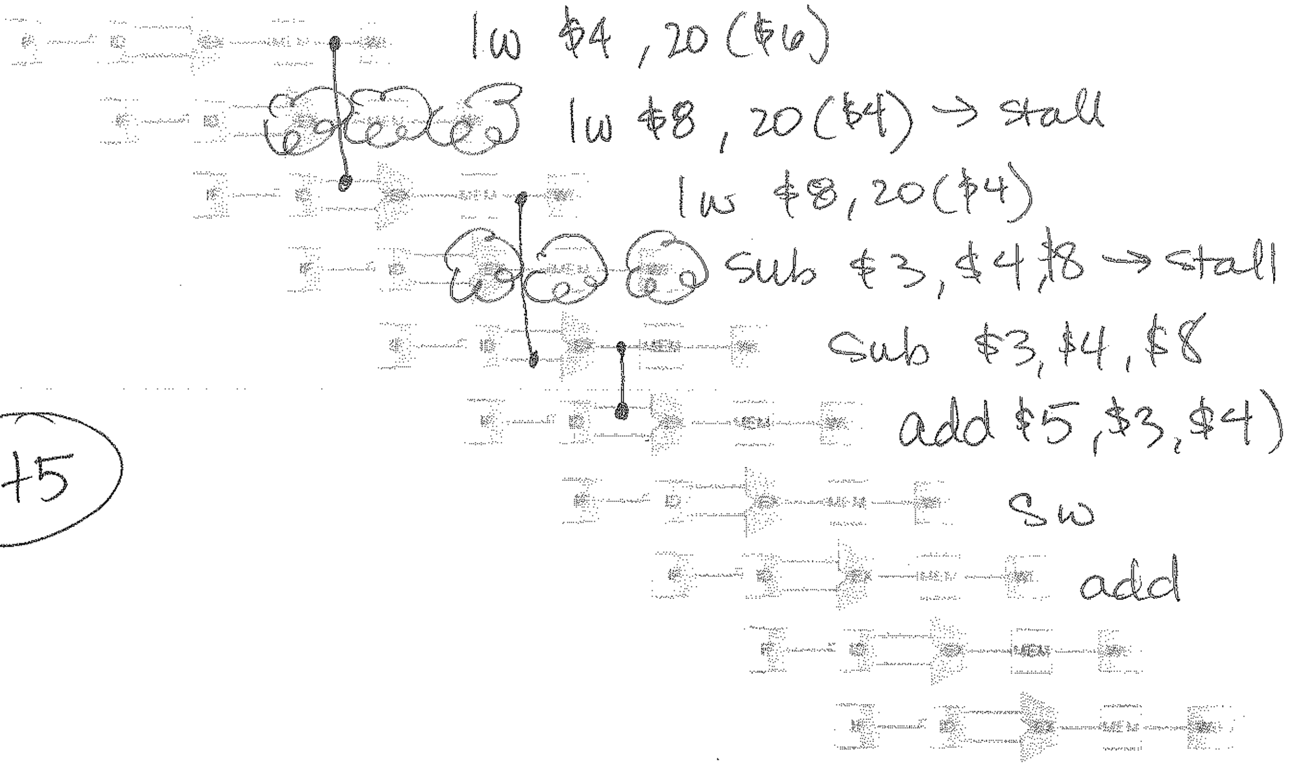
+2 per set of stall

(6 pts)

(more on back)

```
lw $4, 20 ($6)
lw $8, 20 ($4)
sub $3, $4, $8
add $5, $3, $4
sw  $3, 20 ($6)
add $3, $7, $8
```

(c) Assume there **is forwarding** in the pipeline. Label the instructions; show the forwarding paths and\or stalls needed to execute the code.



+5

(d) Can the code be reordered in order to improve performance? If Yes, Show how. If not, why?

No!

we need to insert an instr between lw/lw or lw/sub to remove stalls. All instr are dependent on the values from prev instr. therefore no instr can be inserted to eliminate stalls.

+3

CSE 320 Spring 2014 *Homework #6 Quiz Version A*

Name: SOLUTION *ID:*

No calculators, notes, or textbooks allowed. Show all your work for full credit.
Time limit: 15 mins

Consider the following MIPS code

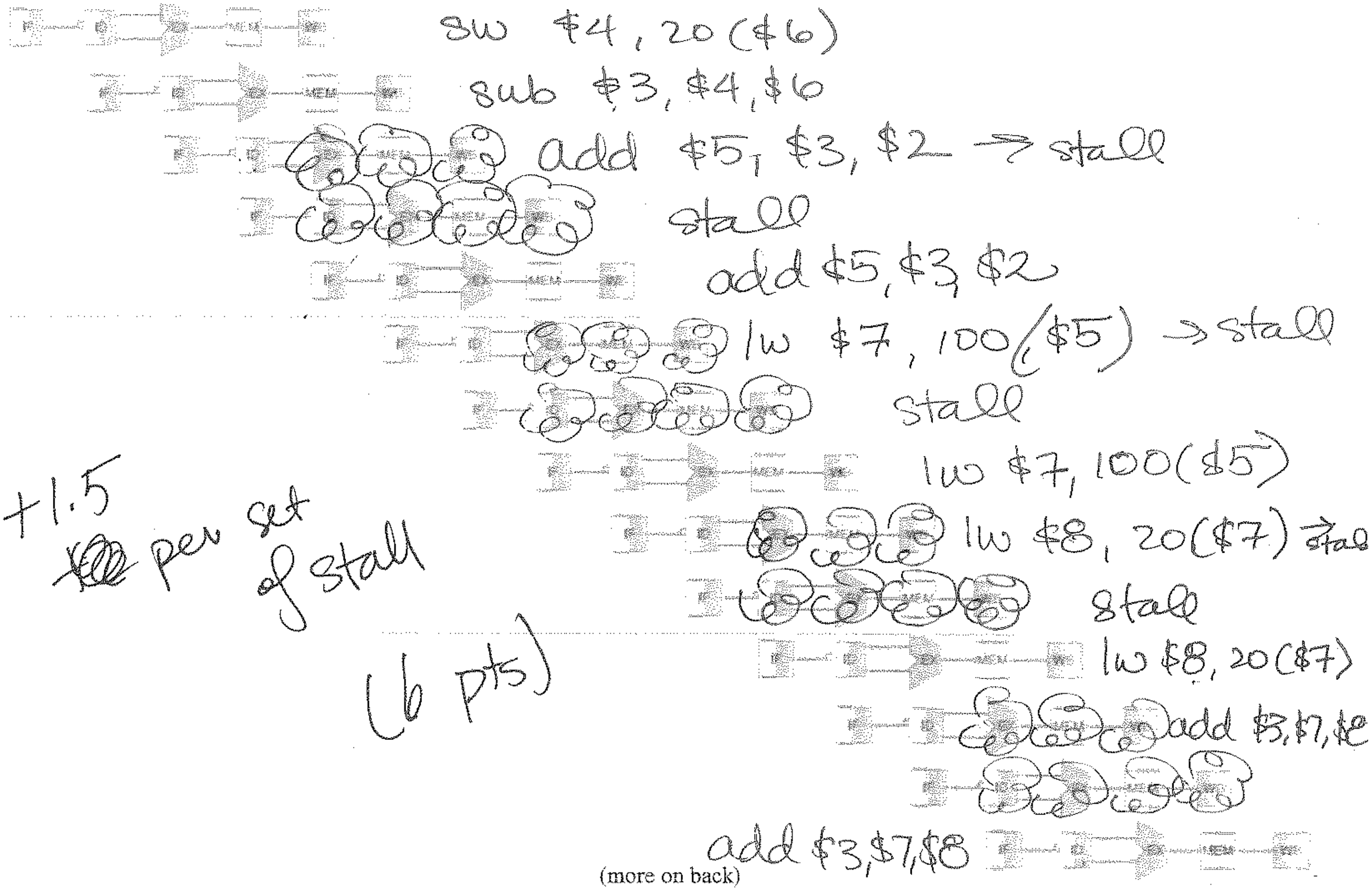
```
sw  $4, 20 ($6)
sub $3, $4, $6
add $5, $3, $2
lw  $7, 100($5)
lw  $8, 20($7)
add $3, $7, $8
```

(a) What type of hazards occurs in the code above? Where do they occur?

+2 data hazards

sub/add $3
lw/add $5
lw/lw $7
add/lw $7 (not actual hazard if stall)
lw/add $8

(b) Assume, there is **no forwarding** in the pipeline. Label the instructions and insert stalls (bubbles) to eliminate all hazards (ie. There should be no forwarding lines)



sw $4, 20($6)

sub $3, $4, $6

add $5, $3, $2 → stall

stall

add $5, $3, $2

lw $7, 100($5) → stall

stall

lw $7, 100($5)

lw $8, 20($7) → stall

stall

lw $8, 20($7)

add $3, $7, $8

add $3, $7, $8

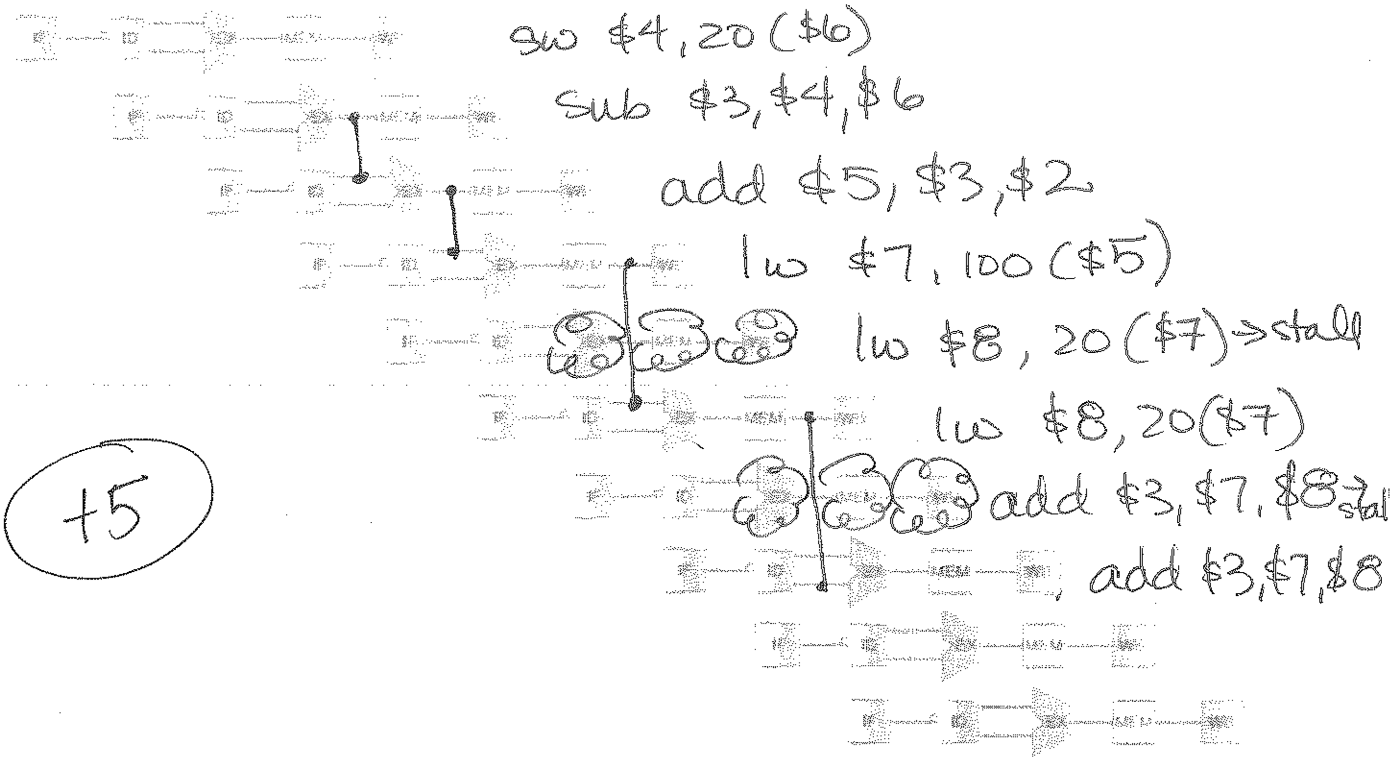
+1.5 ~~[illegible]~~ per set of stall

(6 pts)

(more on back)

```
sw $4, 20 ($6)
sub $3, $4, $6
add $5, $3, $2
lw $7, 100 ($5)
lw $8, 20 ($7)
add $3, $7, $8
```

(c) Assume there **is forwarding** in the pipeline. Label the instructions; show the forwarding paths and\or stalls needed to execute the code.



+5

(d) Can the code be reordered in order to improve performance? If Yes, Show how. If not, why?

```
sub $3, $4, $6
add , $5, $3 $2
lw  $7, 100 ($5)
sw  $4, 20 ($6)
lw  $8, 20 ($7)
add $3, $7, 8
```

or

```
sub $3, $4, $6
add $5, $3, $2
lw  $7, 100 ($5)
lw  $8, 20 ($7)
sw  $4, 20 ($6)
add $3, $7, $8
```

Yes

+3