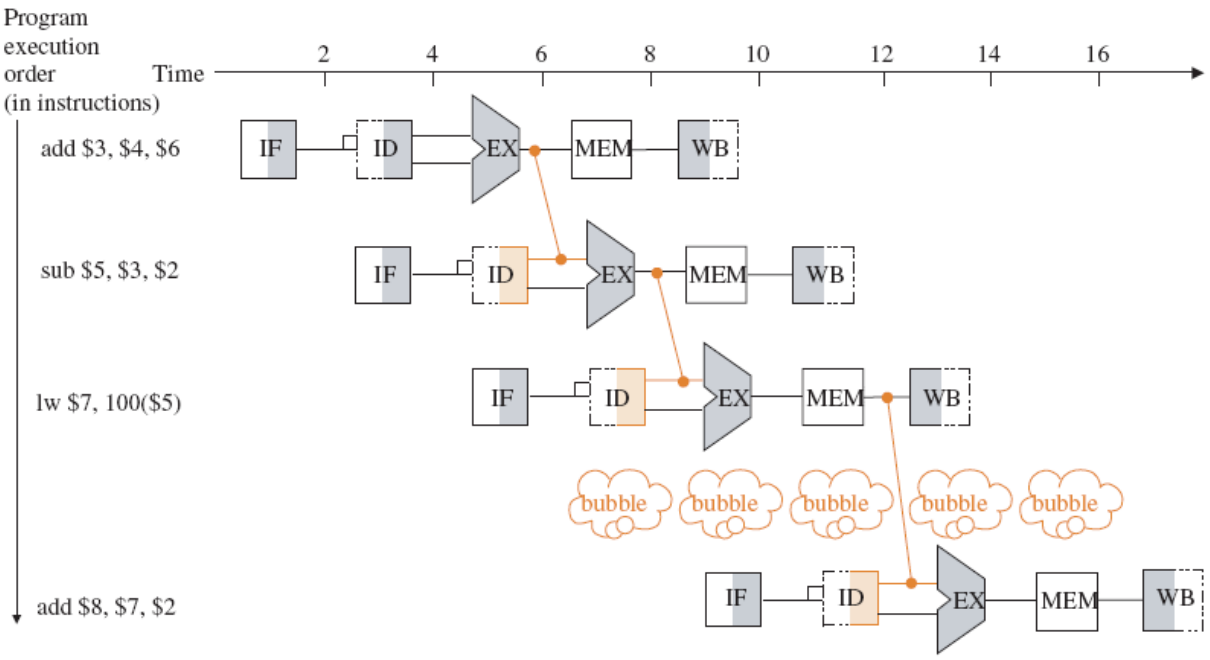
C335 Homework #7 Solution

Part I (10 points)

Using a drawing similar to what we used in lecture notes, show the forwarding paths needed to execute the following four instructions:

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

Solution:



Part II (10 points)

Identify all of the data dependencies in the following code. Which dependencies are data hazards that will be resolved via forwarding? Which dependencies are data hazards that will cause a stall?

```
add   $3, $4, $2
sub   $5, $3, $1
lw    $6, 200($3)
add   $7, $3, $6
```

Solution:

There is a data dependency through $3 between the first instruction and each subsequent instruction. There is a data dependency through $6 between the lw instruction and the last instruction.

For a five-stage pipeline, the data dependency between the first and the second instructions, and the data dependency between the first and the third instructions, are data hazards that can be resolved by using forwarding. The data dependency between the load and the last add instruction cannot be resolved by only using forwarding. A stall is needed.

Part III (10 points)

Here is a series of address references given as word addresses: 2, 3, 11, 16, 21, 13, 64, 48, 19, 11, 3, 22, 4, 27, 6, and 11. Assuming a direct-mapped cache with 16 **one-word** blocks that is initially empty, label each reference in the list as a hit or a miss and show the final contents of the cache.

Solution:

2 (00000010), 3 (00000011), 11 (00001011), 16 (00010000), 21 (00010101), 13 (00001101), 64 (01000000), 48 (00110000), 19 (00010011), 11 (00001011), 3 (00000011), 22 (00010110), 4 (00000100), 27 (00011011), 6 (00000110), 11(00001011)

Tag	Index	Data(one-word block)
0001 → 0100→ 0011	0000	[16] → [64] → **[48]**
	0001	
0000	0010	[2]
0000 → 0001 → 0000	0011	[3] → [19] → **[3]**
0000	0100	**[4]**
0001	0101	**[21]**
0001 → 0000	0110	[22] → **[6]**
	0111	
	1000	
	1001	
	1010	
0000 → 0001 → 0000	1011	[11] → [27] → **[11]**
	1100	
0000	1101	**[13]**
	1110	
	1111	

2 → miss, 3 → miss, 11 → miss, 16 → miss, 21 → miss, 13 → miss, 64 → miss, 48 → miss, 19 → miss, 11 → hit, 3 → miss, 22 → miss, 4 → miss, 27 → miss, 6 → miss, 11 → miss

Part IV (10 points)

Here is a series of address references given as word addresses: 2, 3, 11, 16, 21, 13, 64, 48, 19, 11, 3, 22, 4, 27, 6, and 11. Show the hits and misses and final cache contents for a direct-mapped cache with **four-word** blocks and **a total size of 16 words**.

Solution:

2 (00000010), 3 (00000011), 11 (00001011), 16 (00010000), 21 (00010101), 13 (00001101), 64 (01000000), 48 (00110000), 19 (00010011), 11 (00001011), 3 (00000011), 22 (00010110), 4 (00000100), 27 (00011011), 6 (00000110), 11(00001011)

Tag	Index	Data(four-word block)
0000 → 0001 → 0100 → 0011 → 0001 → 0000	00	[0,1,2,3] → [16, 17, 18, 19] → [64, 65, 66, 67] → [48, 49, 50, 51] → [16, 17, 18, 19] → **[0,1,2,3]**
0001 → 0000	01	[20, 21, 22, 23] → **[4,5,6,7]**
0000 → 0001 → 0000	10	[8,9,10,11] → [24, 25, 26, 27] → **[8,9,10,11]**
0000	11	**[12, 13, 14 ,15]**

2 → miss, 3 → hit, 11 → miss, 16 → miss, 21 → miss, 13 → miss, 64 → miss, 48 → miss, 19 → miss, 11 → hit, 3 → miss, 22 → hit, 4 → miss, 27 → miss, 6 → hit, 11 → miss