1.

(a) Possible buffer in L1:

L1: Write-back buffer

Possible buffer in L2

L2: Write buffer

(b) Possible buffer in L1

L1: Write back buffer

Possible buffer in L2

L2: Write buffer

(a) Procedure:

1. Allocate cache block for the missing data, select a replacement victim.

2. If victim dirty, put it into the write-back buffer, which will be further forwarded into L2 writebuffer

3. Issue write miss request to the L2 cache

4. if hit in L2, source data into L1 cache; if miss, send write request to memory

5. Data arrives and is installed in L1 cache

6. Processor resumes execution and hits, in L1 cache, set the dirty bit.

(b) 1) If L1 miss, allocate cache block for the missing data, select a replacement victim

2) If victim dirty, put it into the write-back buffer, which will be further forwarded into L2 write buffer

3) Issue write miss request to the L2 cache

4) If hit in L2, source data into L1 cache, go to (8)

5) if miss, send write request to memory.

6) Data arrives and is installed in L2 cache

7) Data arrives and is installed in L1 cache

8) Processor resumes execution and hits in L1 cache, set the dirty bit.

3 for chap. 5.5

1)

a) Procedure:

1) Allocate cache block for the missing data, select a replacement victim

2) If victim clean, put it into a victim buffer between both L1 and L2 caches

3) If victim dirty put it into the write-back buffer, which will be further forwarded into L2 write buffer

4) Issue write miss request to the L2 cache

5) If hit in L2, source data into L2 cache, invalidate the L2 copy.

6) Data arrives and is installed in L1 cache

7) Processor resumes execution and hits in L1 cache, set the dirty bit

② Procedure:

1) if L1 miss, allocate cache block for the missing data, select a replacement victim.

2) if victim clean put it into victim buffer between L1 and L2
   if victim dirty put it into the write back buffer, which will be further forwarded into L2 write buffer

3) Issue write miss request to the L2 cache

4) If hit in L2, source data into L1 cache, invalidate copy in L2. Go to (8)

5) if miss, send write request to memory

6) Data arrives and is installed in L2 cache
   in L1 cache

7)

8) Processor resumes execution and hits in L1 cache, set the dirty bit