### Question 1 (55 points) – Prof. Zohar

**A.**

\[
\text{CPI} = (\text{Ideal CPI}) + P_{\text{mem-access}} \times (\text{miss rate}) \times \left[ (\text{addr. translation for new line}) + (\text{allocate new line in cache}) + (\text{dirty replacement}) + (\text{addr. translation for dirty block}) \right]
\]

\[
= 1.36 + 1.3 \times 0.1 \times [20 \times 4 + (24+32/4) \times 4 + 0.4 \times (24+32/4) \times 4 + 0.4 \times 20 \times 4] = 39.2.
\]

Notes:
- Many forgot to include 100% code-reads in addition to 20% loads + 10% stores.
- In virtual addr. cache – we need translation on dirty block write-backs also.
- Any operation cannot take less than CPI-ideal cycles (a LOAD with cache hit will take 1.36 cycles and not 1).

**B.**

\[
\frac{\text{BW-in use}}{\text{BW-max}} = \left( \frac{\text{CPU-freq}}{\text{CPI}} \times (\# \text{line transfers per instruction}) \times (\text{line-size}) \right) / \left[ \frac{\text{Mem-freq} \times (\text{bytes per cycle})}{[800 \times 39.2] \times (130\% \times 10\% \times 1.4) \times 32} / [200 \times 4] \right] = 0.149.
\]

Notes:
- The BW-max can also be regarded as the bandwidth achieved by constantly transferring lines:
  \((24+8)/32 \times 200\text{Mhz}=200\text{Mhz}.

**C.**

\[
\text{CPI} = (\text{Ideal CPI}) + P_{\text{mem-access}} \times (\text{miss rate}) \times \left[ (\text{allocate new line in cache}) + (\text{dirty replacement}) + P_{\text{TLB-miss}} \times (\text{addr. translation for new line}) \right]
\]

\[
= 1.36 + 1.3 \times [0.1 \times (24+32/4) \times 4 + 0.4 \times (24+32/4) \times 4 + 0.005 \times 20 \times 4] = 25.06
\]

Notes:
- TLB miss penalties should be calculated on all memory accesses (regardless of hit/miss).

### Question 2 (55 points) – Prof. Zohar

**Inclusive vs. exclusive L2 cache – (5 points)**

**Inclusive cache**

1. L1 hit – move data from L1 to CPU
2. L1 miss L2 hit – if dirty evict a block from L1 (write back to L2 only), copy block from L2 to L1 and CPU
3. L1 miss L2 miss – evict a block from L1, write back to L2 if it’s dirty. evict a block from L2, write back to main mem if its dirty. Bring a block from mem to L2, and a block (not necessarily the same one) to L1 & CPU.

**Exclusive cache**

1. L1 hit – move data from L1 to CPU
2. L1 miss L2 hit – evict a block from L1 to L2, move block from L2 to L1 and CPU. If the blocks can be mapped to the same set in L2, a simple swap is made, if not, another block from L2 maybe evicted (possibly written back) to main mem.
3. L1 miss L2 miss – evict a block from L1 to L2, evict a block from L2 to main mem, bring a block from main mem to L1 & CPU. The write back operations maybe executed after the needed data is expedited to L1 & CPU.

In this case, when there is L1 miss and L2 hit, it is always possible to swap the two blocks between L1 and L2, and nothing is thrown out of L2.

Using the single cache table, it is possible to estimate the L1 MR and the total MR. The Local L2 MR can then be found by:

\[
\text{HR(total)} = \text{HR(L1)} + (1-\text{HR(L1)}) \times \text{HR(L2)}
\]

\[
\text{HR(L2)} = \frac{\text{HR(total)} - \text{HR(L1)}}{1 - \text{HR(L1)}}
\]

\[
\text{MR(L2)} = 1 - (\text{MR(L1)} - \text{MR(total)}) / \text{MR(L1)}
\]
For the inclusive system, MR(L1 @ 4K) = 7.5%, MR(total @ 16K) = 3%, -> MR(L2) = 40%
For the exclusive system, MR(L1 @ 4K) = 7.5%, MR(total @ 20K) = 2%, -> MR(L2) = 26.66%

The AMAT is then:
AMAT = L1-Hit-time + MR(L1)*(L2-Hit-time + MR(L2)*Mem-line-fill)

AMAT(inc) = 1 + 7.5% * (4 + 40% * 20) = 1.9
AMAT(exc) = 1+ 7.5% * (4 + 26.66% * 20) = 1.7

There are a few options that fit the area:
 a) L2 - 16K, L1 - 4K or smaller
 b) L2 - 8K, L1 - 8K or smaller
 for inclusive cache option (a) is clearly better, but for exclusive cache, one must also verify that:
For the exclusive system, MR(L1 @ 8K) = 4.5%, MR(total @ 16K) = 3%, -> MR(L2) = 66.66%

AMAT(exc) = 1+ 4.5% * (4 + 66.66% * 20) = 1.78

So option (a) prevails for both cases.