1. In order to maximize the chance that the next write or read is a hit - write-allocate puts data from that location in the cache, so if subsequent reads or writes are to the same or nearby blocks, these blocks are already in the cache.

2. $T_{avg} = (1 - L_1 \text{ hit}) \cdot T_{L1} + (1 - L_1 \text{ miss}) \cdot T_{L2} = 0.9 \cdot 1\text{ns} + 0.1 \cdot 10\text{ns} = 1.9\text{ ns}$

3a. 64KB pages ⇒ 16-bit POFS

64 bit VA - 16 bit POFS = 48 bit VPN ⇒ 2^{48} virtual pages

b. 4GB physical memory ⇒ 32-bit PA

32-bit PA - 16-bit POFS = 16-bit PPN ⇒ 2^{16} physical pages

c. 48 bits of VPN to 16 bits of PPN

d. 16-bit PPN ⇒ 16-bit (2B) PTE $\quad 2^6 \cdot 2^0 \cdot \text{byte} \;/\; 2^1 \text{ byte}$

e. 64 KB pages / 2B per PTE = 32K PTEs per page

f. 64-bit VA ⇒ 64-bit (8B) pointers

64 KB pages / 8B pointers ⇒ 8K pointers per page

g. 2^{48} VPs · 2B PTEs = 2^{49} B page table (insanely huge - flat page table is impossible)

h. 1) POFS bits are unaffected by translation; they are the same for virtual and physical
2) VAs are 64-bit with the 16 LSBs reserved for POFS; $35012 < 2^{16}$, so the POFS is 35012_{10}

$35012_{10} \rightarrow 1000100011 0001_2$ which are the POFS bits for virtual and physical mem.

i. No - a page fault only occurs if a page is not in memory. The TLB is smaller than main memory, so a page miss in the TLB may be a hit in memory and not a page fault.