1. (6 points) Complete the state of the NUMA control structures for the following access sequence. Time progresses first by rows.

- **Initial State**
  - CPU 0: Cache
  - CPU 1: Cache
  - CPU 2: Cache
  - Directory: 
    - Directory Entry: 0 1 2
    - X: U 0 0 0
  - Memory: X=2

- P1 reads X
  - CPU 0: Cache
  - CPU 1: Cache
  - CPU 2: Cache
  - Directory: 
    - Directory Entry: 
      - X: U 0 0 0
  - Memory: 
  - CPU 0: 
  - CPU 1: 
  - CPU 2: 

- P2 reads X
  - CPU 0: Cache
  - CPU 1: Cache
  - CPU 2: Cache
  - Directory: 
    - Directory Entry: 
      - X: U 0 0 0
  - Memory: 
  - CPU 0: 
  - CPU 1: 
  - CPU 2: 

- P1 writes X
  - CPU 0: Cache
  - CPU 1: Cache
  - CPU 2: Cache
  - Directory: 
    - Directory Entry: 
      - X: U 0 0 0
  - Memory: 
  - CPU 0: 
  - CPU 1: 
  - CPU 2: 

- P0 reads X
  - CPU 0: Cache
  - CPU 1: Cache
  - CPU 2: Cache
  - Directory: 
    - Directory Entry: 
      - X: U 0 0 0
  - Memory: 
  - CPU 0: 
  - CPU 1: 
  - CPU 2: 

- P2 reads X
  - CPU 0: Cache
  - CPU 1: Cache
  - CPU 2: Cache
  - Directory: 
    - Directory Entry: 
      - X: U 0 0 0
  - Memory: 
  - CPU 0: 
  - CPU 1: 
  - CPU 2: 

X=2

0 1 2
X: U 0 0 0

0 0 0
2. (6 points) Remember that handheld game system from the first few homeworks and quizzes? They’ve got a game running on it now, but it’s way too slow. They want the game to run at **30 frames per second**, but currently it’s taking **50ms per frame**. A quick profiling shows it spends it’s time as follows:

- 40% Graphics
- 15% Sound
- 20% AI
- 20% world update
- 5% input handling

Seeing how much time the game is spending on graphics, management tells everyone to stop what they’re working on and optimize the graphics code.

a. What speedup in graphics alone would achieve the desired frame rate?

b. Assuming that the graphics code was written by reasonably competent engineers, is this speedup likely to be achievable? Were the manager’s instructions appropriate?

3. (8 points) Diagram: A virtual address, a physical address, a cache line, a TLB entry, a cache lookups, and a TLB lookup for the following memory system:

- 512MB physical address space
- 32-bit virtual address
- 16KB Pages
- 256-entry 2-way set-associative TLB
- 24KB 3-way set associative 1024 line cache