CS503 Spring 2019 - Quiz 3B (10 pts)

Answer the questions in the spaces provided. If you run out of room, continue on the back.
Provide clear answers and justify statements where required.
Unless otherwise noted, assume that questions refer to C code, AT&T assembly syntax and the x86 architecture.

1. (2 points) Name:

2. (6 points) State whether the following statements are true of false. **Explain the false statements.**

   (a) (1 point) Xinu has a high-level memory manager and a low-level memory manager.
   - ◯ True
   - ◯ False. Explanation:

   Solution: True.

   (b) (1 point) In Xinu a process cannot kill itself.
   - ◯ True
   - ◯ False. Explanation:

   Solution: False. A process can kill itself.

   (c) (1 point) In x86, when paging is enabled, page tables are stored in a special register (CR3).
   - ◯ True
   - ◯ False. Explanation:

   Solution: False. Page tables are store in memory.

   (d) (1 point) The TLB can speed up data accesses but not instruction fetches.
   - ◯ True
   - ◯ False. Explanation:

   Solution: False. Can speed up both.

   (e) (1 point) Paging divides the logical address space into blocks called page frames.
   - ◯ True
   - ◯ False. Explanation:

   Solution: False. Logical address space gets divided into pages.

   (f) (1 point) With 32-bit addresses and 4KB pages, the 22 most significant bits of the address identify the page number.
   - ◯ True
   - ◯ False. Explanation:

Solution: False. 20 most significant bits.

3. (2 points) Provide definitions for internal and external fragmentation. Draw a picture show internal fragmentation in a pure paging system (not paged segmentation or segmented paging, but only paging). Draw a picture showing external fragmentation in a pure segmentation system.

Solution: Internal fragmentation: space wasted due to fixed block allocation. External fragmentation: space wasted due to contiguous allocations.

4. (2 points (bonus)) Some architectures, like x86, support different page sizes (e.g., 2MB vs. 4KB). For such architectures, it is up to the kernel developer to decide which page size to use. List two reasons why it may be preferable to use small pages. List two other reasons why it may be preferable to use large pages.

Solution: Large pages: Fewer TLB entries used; less space used with page tables. Small pages: less internal fragmentation, lower swapping latency.