

CSSE 332 -- OPERATING SYSTEMS

Multi-level Page Tables

Name:

SOLUTION KEY

Question 1. (10 points) Based on our discussion of a two-level page table with 16-bits addresses and 16-bit PTEs, draw how the following address would be used to lookup the corresponding physical address: `0x3D0B`.

Make sure to show your offset inside each page. Assume that the address of the first level page table is already provided in the appropriate register.

Solution: Solution provided in class.

Question 2. (5 points) Describe the RISC-V organization of a 64-bit address to support multi-level page tables. Make sure to label each section of the address with its use in the address translation.

6362616059585756555453525150494847464544434241403938373635343332313029282726252423222120191817161514131211109876543210

Unused	Level 2 addr	Level 1 addr	Level 0 addr	Page offset
--------	--------------	--------------	--------------	-------------

Question 3. (5 points) In RISC-V, the address of the first level page table for every running process is stored in the satp register.

Question 4. Assume we are dealing with 4 KB pages in RISC-V with the address breakdown from **Question 2**. Answer the following questions.

(a) (5 points) How many page table entries (PTEs) does each page of the page table contain?

Solution:

Each PTE address is 9 bits, which means we have $2^9 = 512$ PTEs.

(b) (5 points) Given that, how wide is a PTE?

Solution: We have 4 KB pages with 512 PTEs per page. This means we have $\frac{2^2 \times 2^{10}}{2^9} = 2^3 = 8 \text{ bytes} = 64 \text{ bits}$.

(c) (5 points) Describe the breakdown of a PTE into its corresponding constituents.

Solution:

63 55 16 05 95 55 65 55 55 55 50 19 81 71 61 51 43 21 10 88 83 63 53 43 33 23 02 22 22 22 22 22 01 91 81 71 61 51 43 21 10 9 8 7 6 5 4 3 2 1 0

Unused

Frame Address

Flags