

HW9.1. Virtual Memory

Q1. We want to set up a virtual memory system with **4 GiB of virtual memory**, mapped to **4 MiB of physical memory**.

Q1.1: How many bits long will our virtual addresses be?
Given the size of the virtual memory, how many bits do we need to address each byte of the virtual memory?

integer

?

Q1.2: How many bits long will our physical addresses be?
Given the size of the physical memory, how many bits do we need to address each byte of the physical memory?

integer

?

Q2. In order to set this up, we decide to use a **page size of 4 KiB** (4096 bytes), with a single layer page table.

Q2.1: How many bits long is the page offset?
The page offset addresses each byte of the page. Given the page size, how many bits do we need to address each byte of the page?

integer

?

Q2.2: How many bits long is the virtual page number (VPN)?
The virtual address is partitioned into the virtual page number and the page offset.

integer

?

Q2.3: How many bits long is the physical page number (PPN)?
The physical address is partitioned into the physical page number and the page offset.

integer

?

Q3. We decide to include metadata and padding for each page table entry such that **each entry in our page table is exactly 32 bits (4 bytes) long**. For the purposes of this class, we assume that our page table is structured like an array of entries, with a slot for each virtual address.

Q3.1: How many page table entries are required for the full page table?
The VPN is the index of the page table. Given the VPN bits computed earlier, how many entries would the page table have?

integer

?

Q3.2: How many bytes of data does the page table take?
Knowing the number of page table entries and the number of bytes for each page table entry, how many bytes is the whole page table?

integer

?

Q3.3: How many pages of data does the page table take?
Given the size of each page and the computed page table size, how many pages would the page table consume?

integer

?

Q3.4: How many physical pages can we use for actual data, if we decide to use this page table?

Homework 9

Assessment overview

Total points:

0/100

Score:

0%

Question

Value:

20

History:

Awarded points:

0/20

Report an error in this question

Previous question

Next question

Attached files

No attached files

Attach a file

Attach text

Given the size allocated for page tables (page tables are also stored in the physical memory) and the actual size of physical memory, how much space do we have left for the actual data?

integer

?

Q3.5: With a one-level page table, how many physical memory accesses are required per virtual memory access?

The page table (stored in physical memory) contains the address translation to where the actual data is located (also in physical memory).

integer

?

Q4. In order to make things more efficient, we decide to use a two-level page table instead, with the **virtual page number bits split evenly** between the two layers. We still use a page table entry size of 32 bits (4 bytes).

Q4.1: How many page table entries are in the L1 page table?

VPN1 indexes the L1 page table. How many bits do we have for VPN1? How many entries would that translate to?

integer

?

Q4.2: How many pages does our L1 page table use?

Given the size of each page table entry and the computed number of entries for L1, how many bytes does the L1 page table consume? Given the page size, how many pages would the L1 page table consume?

integer

?

Q4.3: How many page table entries are in each L2 page table?

VPN2 indexes the L2 page table. How many bits do we have for VPN2? How many entries would that translate to?

integer

?

Q4.4: How many pages does each L2 page table use?

Given the size of each page table entry and the computed number of entries for L2, how many bytes does the L2 page table consume? Given the page size, how many pages would the L2 page table consume?

integer

?

Q4.5: With a two-level page table, how many physical memory accesses are required per virtual memory access?

The L1 page table will point you to the L2 page table. The L2 page table will point you to the actual translation. Both of the page tables are stored in physical memory.

integer

?

Save & Grade 20 attempts left

Save only

Additional attempts available with new variants ?