Coral S. Schmidt Montilla #148830

1. Consider a pure paging system that uses 32-bit addresses (each of which specifies one byte of memory), contains 128MB of main memory and has a page size of 8KB.
2. How many pages frames does the system contains?



1. How many bits does the system use to maintain the displacement, d?



1. How many bits does the system use to maintain the page number, p?



1. Consider a pure paging system that uses three levels of page tables and 64-bit addresses. Each virtual address is the ordered set v = (p, m, t, d), where the ordered triple (p, m, t) is the page number and d is the displacement in to the page. Each page table entry is 64 bits (8 bytes). The number of bits that store p is np, the number of bits that store m is mn and the number of bits to store t is nt.
2. Assume np = nm = nt = 18
   1. How large is the table at each level of the multilevel table?



* 1. What is the page size, in bytes?



1. Assume np = nm = nt = 14.
   1. How large the table at each level of the multilevel page table?



* 1. What is the page size, in bytes?



1. Discuss the trade-off of large and small sizes.

The trade-off between large and small sizes in multilevel page tables lies in the balance between each page table's size and the page size's granularity. Larger page sizes reduce the overhead of managing page tables but may lead to internal fragmentation, where not all memory within a page is utilized efficiently. Conversely, smaller page sizes minimize internal fragmentation but require larger page tables, increasing the memory overhead. Therefore, choosing the optimal size involves considering memory usage patterns, access efficiency, and system performance requirements.