Given a 64-bit x86_64 Architecture where the page size is set at 4096, the number of physical address size in memory is 246, and the virtual memory address size is 248, derive the number of pages in the disk and draw the representation of the Virtual-to-Physical address translation.

For a single level Page Table with 32-Bit machine with 4-KB pages, define the size of a single Page Table size. Then, what is the smallest amount of page table data we need to keep in memory for n-32 bit applications in a two-level Hierarchical Page Table? And how many applications can be implemented in this two-level Hierarchical Page Table in the equivalent memory space?

Given a 64-bit x86_64 Architecture where the page size is set at 4KB, the number of physical address size in memory is 246, and the virtual memory address size is 248, derive the number of pages in the disk and draw the representation of the Virtual-to-Physical address translation.

For a single level Page Table with 64-Bit define the size of a single Page Table size. Then, what is the smallest amount of page table data we need to keep in memory for n-32 bit applications in a two-level Hierarchical Page Table? Draw the representation of the Virtual-to-Physical address translation.

Finally, how many applications can be implemented in this two-level Hierarchical Page Table in the equivalent memory space?