Design Document - Operating Systems - Anshul Sharma
==========

For this machine problem, we are asked to implement a Virtual Memory Manager capable of supporting large address spaces and a virtual memory allocator. Note that for this problem we also implement a recursive page table lookup.
For implementing this we extend the page_table.c file and also implement vm_pool.c file.

Here's the list of functions implemented:
==========

1. Pagetable(): in the constructor we include the recursive setting of the page table where the last entry(1023) points to itself, apart from the earlier implementation(mp3) of this function.
2. load(): here in the page table base register(CR3), we load the current object's page directory index.
3. enable_paging(): use it to set CR0 register to signify that the paging is enabled.
4. handle_fault(): in this api we first find the faulty address and then a free frame is found from the process frame pool to map the address in the page table and directory.