Xi Liu, xl3504

1.1
In physical memory, kernel's code and data live in 0x040000

1.2
In physical memory, kernel's stack lives on the page that is one page lower in address than 0x80000
0x80000 in binary is 0b1000 0000 0000 0000 0000
0x80000 in decimal is 2^19 = 5242
Page size is 2^12 = 4096
5242 – 4096 = 1146
1146 in decimal is 0x47A in hexadecimal
So the kernel's stack live on 0x47A

1.3
In virtual memory, kernel's code and data live in 0x040000
In virtual memory, kernel's stack lives on 0x47A

1.4
Number of physical pages in memory = MEMSIZE_PHYSICAL / PAGESIZE

2.1
Pages are 4KB
The process allocates 12KB
Number of pages for allocated memory: 12KB / 4KB = 3 pages
Page table structure has 4 levels, each level has 1 page
Total: 3 + 4 = 7 pages

2.2
One L4 page table can hold 2^9 entries
2^9 + 2^9 > 2^9 + 1 > 2^9, so two L4 page tables are needed
Number of pages for allocated memory: 2^9 + 1 pages
Page structure: L1 has 1 page, L2 has 1 page, L3 has 1 page, L4 has 2 pages
Total: (2^9 + 1) + (1 + 1 + 1 + 2) = 2^9 + 6

2.3
One L3 page table can hold 2^9 * 2^9 = 2^{18} entries
2^{18} + 2^{18} > 2^{18} + 1 > 2^{18}, so two L3 page tables are needed
Number of pages for allocated memory: 2^{18} + 1 pages
Page structure: L1 has 1 page, L2 has 1 page, L3 has 2 pages, L4 has 2^9 + 1 pages
(One L3 page table entry contains the base address of the L4 table that has 2^9 page table entries; another L3 page table entry contains the base address of the L4 table that

has 1 page table entry)
Total: (2^{18} + 1) + (1 + 1 + 2 + 2^9 + 1) = 2^{18} + 2^9 + 6

3.1
Sometimes
A page fault does not imply a TLB miss. When a page is read-only, and user-level process attempts to write to it. TLB is caches the mapping. However, this would be a permission violation, which would cause a page fault.
If the page is on disk and not cached in TLB, then the memory reference would cause both a TLB miss and a page fault

3.2
Sometimes
When TLB is flushed after context switches, the memory reference would cause a TLB miss, but the mapping is in main physical memory, so there would not be a page fault

3.3
Sometimes
Same as above, being in user mode or kernel mode does not influence the possibility of getting a page fault

3.4
Always
Regardless of whether the R/W bit is set to read-only or read/write access permission, the process has the ability to load (read) once the page table's entry PTE_P and PTE_U bits are set

3.5
Sometimes
If the R/W bit is set to ready-only, then the process cannot store (write) to the page table entry