(1) What is “virtual memory”?

The virtual memory is a mechanism (or “a method”) that expands memory space by mapping some of the virtual memory address space to hard drive.

(2) What is “page fault” (make sure to answer this question using the appropriate technical term(s) the CS286 lectures already covered)?

Page fault is a situation where a virtual memory page requested by a processor is not mapped to any physical memory page.

(3) How is “dirty flag” used in virtual memory for (also explain its primary purpose)?

The dirty flag in virtual memory is used for eliminating one (possibly) unnecessary hard disk access during a page fault. The dirty flag can eliminate unnecessary hard disk access when a dirty flag is ‘0’ (when the virtual memory page is being kicked out of the physical memory page). It (a dirty flag = 0) indicates that a virtual memory page in a physical memory page has never been updated ever since the virtual memory page was loaded to the physical memory page (otherwise, the virtual memory page needs to be saved back to the hard drive).
(4) What are the primary advantages in using “virtual memory (using demand paging)”?
Mention at least three different advantages in using “virtual memory”.

- Programs (or data sets) that are larger than the physical memory can be executed (or loaded to the memory).
- Large programs will be started quicker.
- More programs (than the capacity of the physical memory) can be active at a time.

(5) How is “segmentation fault” caused (make sure to answer this question using the appropriate technical term(s) the CS286 lectures already covered)?

Segmentation fault is caused when a processor generates the segment offset address, which exceeds the size of the memory segment the segment offset is for.