#1: Why are “boot sector viruses” are fatal to systems?

#2: How is “MFT” in NTFS different from “FAT” (mention the most significant difference)?

#3: NTFS uses a tree structure instead of a linear linked-list (which is used in FAT) for managing the disk space for each file. What is the major reason why NTFS uses a tree structure instead of a liner linked-list?

#4: What is the primary difference between “B tree” and “B+ tree”? What is the primary reason NTFS uses “B+ tree” (instead of “B tree”)?

#5: What is “virtual memory” and what are the primary advantages in virtual memory (at least three)?

#6: What is “page faults”? Why are page-faults a serious issue?

#7: Which of FIFO or LRU memory page replacement algorithm does result in fewer page faults and why?

#8: What is “locality in memory accesses”? What are the two different types of “locality in memory accesses”?

#9: What is “(memory) working set”? How should it be utilized?

#10: What is the major advantage and disadvantage in using LRU memory page replacement algorithm (compared with using “FIFO page replacement algorithm”)?

#11: What is the major advantage in using NRU (Not Recently Used) page-replacement algorithm?

#12: What is the major advantage in using 2nd-chance page-replacement algorithm?

#13: What is “Belady’s Anomaly”? What property (in page replacement algorithm) guarantees to avoid “Belady’s anomaly”?

#14: What is “memory leak” and how does one happens?

#15: What is “(memory) garbage collection”? What type of memory do they “collect”?

#16: Why is “(memory) garbage collection” a problem?

#17: Why is the primary difference between “(memory) garbage collection (CS314)” and “memory compaction (CS286)”?

#18: What is the primary advantage in using a programming-language that does not perform “automatic garbage collection”?