CS286-Computer Organization & Architecture
Possible Quiz Questions (Quiz #8)
For October 22nd

The following is a list of possible questions for our quiz on October 22nd. Some of the questions will not be asked in the quiz. All the questions that will appear in the quiz will appear exactly as shown below (however, numeric parameters may be changed). The quiz is closed textbook, closed notes and closed neighbors. Note that the questions, which did not appear in this quiz, still may appear in the exams. You will find a solution for these questions during lectures.

#1: Show the memory hierarchy.

#2: What is the trade-off problem in the memory hierarchy?

#3: What is “external memory fragmentation”?

#4: How does “external memory fragmentation” happen?

#5: Why is “memory external fragmentation” a serious problem?

#6: What is “compaction”?

#7: What are the problems in compaction?

#8: Where in the memory hierarchy “virtual memory” exists?

#9: What is “virtual memory”?

#10: What are the two advantages in “virtual memory”?

#11: What is “logical memory address space”?

#12: What is “page fault”?

#13: What is “valid flag” used in virtual memory for?

#14: Sketch the contents in VMT (virtual memory table).

#15: What is the primary problem in virtual memory?

#16: What is “the internal memory fragmentation”?

#17: How is “dirty flag” used in virtual memory for?

#18: What is “demand paging”? What is the primary advantage?
#19: What is “locality in memory reference”? What are the two different types of “locality”?

#20: What does “TLB” stand for?

#21: In the virtual memory (as we discussed in the classroom), how many disk accesses can happen in the worst case?

#22: Sketch the structure of “segmentation table (or “segmentation descriptor table”).”

#23: What is the advantage of using segmentation?

#24: How is “segmentation fault” caused?

#25: Which of “page fault” or “segmentation fault” is fatal?

#26: What is “pipeline memory accesses”? What is “memory interleaving” (make sure to emphasize the difference from “pipeline memory accesses”)?