# Section 8: Virtual Memory

NOTE: Please Review Answers Before Posting

## Question 1

What are the two characteristics of simple paging that lead to, and the fundamental idea defines virtual memory? Hint: Three parts in slides.

### Answer

The two characteristics of simple paging are:

1. Because of the indirection between logical and physical addresses, a process’s pages can be located anywhere in memory and can be moved (relocated).
2. The frames occupied by a process’s pages do not need to be contiguous i.e. they can be spread throughout physical memory.

And the fundamental idea in virtual memory is:

1. Not all of a process’s pages need to be maintained in physical memory for the process to execute. Only those pages in the process’s working set need to be resident.

## Question 2

1. Briefly describe Page Fault.
2. Briefly describe Page Fault Rate (PFR).
3. Briefly describe Thrashing.

### Answer

1. A page fault is the interrupt generated by the memory management unit when the process references a page that is not resident in memory i.e. not in the resident set.
2. The page fault rate is a metric that describes the number of page faults that occur over some time period e.g. PFs per minute.
3. Thrashing occurs when a process’s page fault rate (PFR) rises to a point that the process cannot make progress i.e. each time the process is scheduled for execution (is made runnable) a missing page that the process requires and generates a page fault that blocks its execution.

## Question 3

Note: This question assumes a static resident set size.

1. Briefly describe a process’s Resident Set.
2. What is the result of setting a process’s resident set size too small?
3. What is the result of setting a process’s resident set size too large?

### Answer

The resident set is the set of process pages that are resident in (paged into) memory.

A resident set that is too small will cause an increase in the number of page faults and will cause the process to begin thrashing.

A resident set that is too large wastes memory (frames) by maintaining process pages in memory frames that are no longer being referenced by the process. This has the effect of reducing the overall number of processes that can be maintained in memory.

## Question 4

1. Briefly describe a process’s Working Set.
2. Briefly describe the relationship between a process’s working set and its resident set.
3. Generally, how is a process’s page fault frequency used to adjust a process’s resident set size?

### Answer

1. The Working Set is the set of pages that the process currently needs to execute. Any page in the working set that is not resident must be swapped in.
2. The resident set must include at least the working set. If the resident set is smaller than the process’s working set, the process will begin to thrash.
3. If the resident set is smaller than the working set, the PF Frequency will rise providing an indication that the process’s resident set’s size must be increased. When the resident set contains the working set, the PF Frequency will drop. The OS can monitor the PF Frequency using it as an indicator of how to adjust the process’s resident set size up or down.

## Question 5

1. What three pieces of information is maintained in a page table entry (as described in the slides)?
2. How many page table entries are needed with a 32 bit address and a page size of 8192?
3. What is the technique described in the book, and in class, that reduces the number of page table entries needed to be maintained in memory?

### Answer

1. Mainly each entry maintains the upper N bits of the physical address of the memory frame containing the page. The PT Entry also maintains a ‘Present bit’ that indicates whether the page is resident (1) or must be retrieved from disk (0). The PT entry also maintains a ‘modified bit’ that indicates whether the page has been written to (modified) since it was made resident.
2. A 32 bit address range with a page size of 213 bits permits (32-19) 219 (524288) pages.
3. Dividing the page table into multiple levels (two in the book) greatly reduces the number of PTE that must be maintained in memory with the tradeoff of additional complexity in the design of the memory management unit hardware (Figure 5.8).

## Question 6

What are the two advantages of a large page size described in the slides?

### Answer

The large page size results in fewer page table entries (smaller page tables), a smaller TLB, and overall reduces the amount of processing overhead of memory management on the system.

The large page size increases the efficient the I/O operations that move pages between the swap drive and memory i.e. more information is moved between the swap drive and memory per I/O operation.

## Question 7

Describe the purpose of the Translation Lookaside Buffer.

### Answer

The TLB is a memory cache for page table entries. When the TLB hits, it avoids the additional memory reference needed to retrieve one or more page table entries from memory depending on whether a 1, 2, 3, etc. level page table is being employed. Without a TLB, every virtual memory reference would require two or more memory accesses (i.e. 1, 2, or more page table entries) plus the reference to the target frame.

## Question 8

1. Describe the Operating System’s Fetch Policy.
2. How does the Prepaging Fetch Policy reduce the number of page faults experienced when a process is initially started?

### Answer

The fetch policy determines when a page is made resident i.e. swapped from disk into memory.

NOTE: The following discussion concerning pre-paging only applies to a hard disk drive. There is no seek penalty in reading non-adjacent blocks with a solid state drive.

The Prepaging Fetch Policy will not only swap-in the page being referenced by the page fault, but will swap-in the adjacent page in a single I/O operation. The assumption is that if page i is needed, then the adjacent page i+1 will likely be needed very soon after. By bringing both pages in with the same I/O operation, the time needed to retrieve page i+1 is reduced to almost nothing relative to the time needed to bring in page i. That is, the expense in the I/O operation is moving the disk’s heads to the needed track and waiting for the correct disk sector to move into position. The time needed to retrieve an adjacent sector is almost nil compared to the setting up and retrieving the first.

## Question 9

1. Describe the goal of a Page Replacement Policy.
2. In theory, how does the Least Recently Used page replacement policy select a page to be replaced when a new page must be made resident?
3. Describe how the ‘use bit’ marks pages in the Clock Algorithm. How is the use bit used to select a page for replacement?

### Answer

To reduce the page fault rate by selecting pages for replacement that will not be referenced for the longest amount of time.

The LRU policy selects the page that has not been referenced in the longest amount of time. According to the principle of locality, the LRU page is least likely to be referenced in the near future.

The frame table maintains a use bit (UB) for every memory frame. The UB is an indicator of whether the frame has been referenced in the recent past. The Clock page replacement algorithm scan the frame table until a UB = 0 is located which is then selected for replacement. During this scan, if the frame’s UB = 1, it is set to 0 and the next frame examined. Every time a process references a frame (i.e. references the page in the frame), the UB is set to 1. Therefore, the only frames (pages) that have not been recently referenced will be selected for replacement.

Note: You may be required to explain the clock algorithm in an exam question.

## Question 10

1. Describe the goal of a Load Control Policy.
2. Describe how a process’s Working Set can be used to determine which processes to maintain in a ready state or which to suspend.
3. Describe the L=S Criteria approach to determining how many processes to maintain in a ready state.

### Answer

The load control policy determines the number of processes to make active (i.e. not suspended) with the goal of maximizing processor utilization. Too few processes reduce the chance that a process will be in a ‘ready’ state (not blocked for I/O). Too many processes reduces the resident set sizes of individual processes and so increases process thrashing.

The Working Set method schedules processes for execution whose current working set is contained in its resident set and suspends processes whose working set is not currently resident.

The L=S Criterion method maintains the number of executable processes such that the average time between page faults is equal to the average time needed to swap a page from disk (length of the IO operation). Too high a page fault rate indicates that too many processes are active and some need to be suspended to free up frames for the remaining processes.