Chapter 10

Multiple Choice Questions

1. In demand paging,
A) a page loaded in memory may never be accessed.
B) all pages that a program will access during execution are loaded in memory in the beginning.
C) a page is loaded in memory only when it is needed during execution.
D) a page is loaded in memory just before it is needed.

Ans: C
Feedback: 10.2
Difficulty: Easy

2. An advantage of virtual memory is that
A) a program can be much larger than the size of physical memory.
B) the programmers can concentrate programming the problem instead of worrying about the amount of physical memory available.
C) it provides a way to execute a program that is only partially loaded in memory.
D) All of the above.

Ans: D
Feedback: 10.1
Difficulty: Easy

3. If an instruction modifies several different locations, a page fault can be handled by
A) using temporary registers to hold the values of overwritten locations.
B) loading multiple pages in advance.
C) incorporating special hardware.
D) terminating the process.

Ans: A
Feedback: 10.2.1
Difficulty: Easy

4. A free-frame list
A) is a set of all frames that are filled with all zeros.
B) is a set of all frames that are currently unallocated to any process.
C) is a set of all frames that are currently being shared by at least two processes.
D) is a set of all frames that are used for stack and heap memory.

Ans: B
Feedback: 10.2.2
Difficulty: Easy

5. The most preferred method of swapping a process is
A) to swap using the file system.
B) to copy an entire file to swap space at process startup and then perform demand paging from the swap space.
C) to demand-page from the file system initially but to write the pages to swap space as they are replaced.
D) None of the above.

Ans: C
Feedback: 10.2.3
Difficulty: Medium

6. Anonymous memory of a process refers to
A) the pages not associated with the binary executable file of the process.
B) the pages associated with the binary executable file of the process.
C) the pages associated with the static data of the process.
D) the pages that cannot be swapped out of the physical memory.

Ans: A
Feedback: 10.2.3
Difficulty: Medium

7. _____ is the algorithm implemented on most systems.
A) FIFO
B) Least frequently used
C) Most frequently used
D) LRU

Ans: D
Feedback: 10.4.5
Difficulty: Medium

8. The dirty (modify) bit identifies
A) a page that has been corrupted.
B) a page that needs to be reloaded when accessed.
C) a page that is shared by multiple processes.
D) a page that has been modified since it was loaded.

Ans: D
Feedback: 10.4.1
Difficulty: Medium

9. Suppose we have the following page accesses: 1 2 3 4 2 3 4 1 2 1 1 3 1 4 and that there are three frames within our system. Using the FIFO replacement algorithm, what is the number of page faults for the given reference string?
A) 14
B) 8
C) 13
D) 10

Ans: B
Feedback: 10.4.2
Difficulty: Medium

10. Suppose we have the following page accesses: 1 2 3 4 2 3 4 1 2 1 1 3 1 4 and that there are three frames within our system. Using the FIFO replacement algorithm, what will be the final configuration of the three frames following the execution of the given reference string?
A) 4, 1, 3
B) 3, 1, 4
C) 4, 2, 3
D) 3, 4, 2

Ans: D
Feedback: 10.4.2
Difficulty: Medium

11. Optimal page replacement ____.
A) is the page-replacement algorithm most often implemented
B) is used mostly for comparison with other page-replacement schemes
C) can suffer from Belady's anomaly
D) requires that the system keep track of previously used pages

Ans: B
Feedback: 10.4.3
Difficulty: Medium

12. Given the reference string of page accesses: 1 2 3 4 2 3 4 1 2 1 1 3 1 3 and a system with three page frames, what is the final configuration of the three frames after the OPT algorithm is applied?
A) 1, 3, 4
B) 1, 2, 3
C) 2, 3, 4
D) 1, 2, 1

Ans: B
Feedback: 10.4.3
Difficulty: Medium

13. Given the reference string of page accesses: 1 2 3 4 2 3 4 1 2 1 1 3 1 4 and a system with three page frames, what is the final configuration of the three frames after the LRU algorithm is applied?
A) 1, 3, 4
B) 3, 1, 4
C) 4, 1, 2
D) 1, 2, 3

Ans: B
Feedback: 10.4.4
Difficulty: Medium

14. Stack algorithms are a class of page replacement algorithms that
A) are implemented using stacks.
B) are guaranteed to incur the least number of page faults.
C) do not suffer from Belady's anomaly.
D) are guaranteed to incur no more page faults than FIFO page replacement algorithm.

Ans: C
Feedback: 10.4.4
Difficulty: Medium

15. In Additional-Reference-Bits algorithm,
A) the least-recently used page is identified very efficiently on a page fault.
B) a group of pages that have not been used recently are efficiently identified.
C) a page that has been referenced most recently is identified.
D) None of the above.

Ans: B
Feedback: 10.4.5
Difficulty: Medium

16. The Second-Chance algorithm
A) is same as FIFO algorithm if all pages in memory have been referenced at least once since the last page fault.
B) is same as FIFO algorithm if none of the pages in memory have been referenced since the last page fault.
C) is same as LRU algorithm if all pages in memory have been referenced at least once since the last page fault.
D) is same as LRU algorithm if none of the pages in memory have been referenced since the last page fault.

Ans: A
Feedback: 10.4.5
Difficulty: Medium

17. Counting based page replacement algorithms are not commonly used, because
A) they only approximate LRU.
B) they are expensive to implement, even though they incur the least number of page faults.
C)they do not approximate OPT replacement very well and their implementation is expensive.
D) they do not approximate OPT replacement very well, even though they are easy to implement.

Ans: C
Feedback: 10.4.6
Difficulty: Medium

18. If an architecture has a move instruction with more than one word and at most one of the two operands may be an indirect memory reference, the minimum number of frames needed to run a process on this architecture is
A) 3
B) 4
C) 5
D) 6

Ans: C
Feedback: 10.5.1
Difficulty: Difficult

19. A drawback of equal or proportional allocation is that
A) they are very expensive to compute.
B) the processes that arrive earlier get more pages than the processes arriving later.
C) the allocation varies according to the degree of multiprogramming.
D) a high-priority process is treated the same as a low-priority process.

Ans: D
Feedback: 10.5.2
Difficulty: Medium

20. Which of the following is FALSE about reapers?
A) They may swap out a page from a process even when that process is not running.
B) They may use any page replacement algorithm to swap out pages.
C) They may swap out pages event when there are ample free frames available.
D) They implement a global page replacement policy.

Ans: C
Feedback: 10.5.3
Difficulty: Medium

21. A sign of thrashing is
A) the CPU utilization increases as the degree of multiprogramming is increased.
B) the CPU utilization decreases as the degree of multiprogramming is increased.
C) the CPU utilization increases as the number of pages allocated to each process is increased.
D) the CPU utilization decreases as the number of pages allocated to each process is increased.

Ans: B
Feedback: 10.6.1
Difficulty: Easy

22. The working set strategy
A) swaps out a process if OS cannot allocate enough pages to accommodate its working set.
B) swaps in a new process if there is enough memory available to accommodate its working set.
C) keeps the degree of multiprogramming as high as possible while preventing thrashing.
D) All of the above.

Ans: D
Feedback: 10.6.2
Difficulty: Medium

23. Which of the following is true about the strategy that uses page fault frequency (PFF) to prevent thrashing?
A) A new page is allocated to a process if PFF is too high.
B) A page is deallocated from a process if the PFF is too low.
C) A new process may be swapped in if PFF is too low.
D) All of the above.

Ans: D
Feedback: 10.6.3
Difficulty: Easy

24. Memory compression is a useful alternative to paging
A) when the system has slow magnetic disks but not faster SSDs.
B) when the compression algorithm is can achieve the best possible compression ratio.
C) when the speed of the compression algorithm fastest.
D) even when the system has faster SSDs.

Ans: D
Feedback: 10.7
Difficulty: Medium

25. Which of the following is true about choosing an appropriate page size?
A) Larger page size results in reducing total I/O.
B) Smaller page size reduces the number of page faults.
C) Larger page size reduces I/O time.
D) Larger page size results in less total allocated memory.

Ans: C
Feedback: 10.9.2
Difficulty: Medium

26. Which of the following does not increase the TLB reach?
A) Increase the size of TLB.
B) Increase page size.
C) Decrease the size working set.
D) Provide multiple page sizes.

Ans: C
Feedback: 10.9.3
Difficulty: Medium

27. In which of the following cases, performance may be improved?
A) Designing array access pattern based on whether the arrays are stored in row major or column major order.
B) Placing frequently interacting components close to one another.
C) Avoid placing functions across page boundaries.
D) All of the above.

Ans: D
Feedback: 10.9.5
Difficulty: Medium

28. Lock bit is NOT used for which of the following cases:
A) the page that is being used for I/O.
B) some OS pages.
C) a new page that has just been brought in.
D) a page that belongs to a high priority process.

Ans: D
Feedback: 10.9.6
Difficulty: Medium

Essay Questions

1. What is the benefit of using sparse addresses in virtual memory?

Ans: Virtual address spaces that include holes between the heap and stack are known as sparse address spaces. Using a sparse address space is beneficial because the holes can be filled as the stack or heap segments grow, or when we wish to dynamically link libraries (or possibly other shared objects) during program execution.
Feedback: 10.1
Difficulty: Medium

2. Explain the distinction between a demand-paging system and a paging system with swapping.

Ans: A demand-paging system is similar to a paging system with swapping where processes reside in secondary memory. With demand paging, when a process is executed, it is swapped into memory. Rather than swapping the entire process into memory, however, a lazy swapper is used. A lazy swapper never swaps a page into memory unless that page will be needed. Thus, a paging system with swapping manipulates entire processes, whereas a demand pager is concerned with the individual pages of a process.
Feedback: 10.2
Difficulty: Difficult

3. Explain the sequence of events that happens when a page-fault occurs.

Ans: When the operating system cannot load the desired page into memory, a page-fault occurs. First, the memory reference is checked for validity. In the case of an invalid request, the program will be terminated. If the request was valid, a free frame is located. A disk operation is then scheduled to read the page into the frame just found, update the page table, restart the instruction that was interrupted because of the page fault, and use the page accordingly.
Feedback: 10.2
Difficulty: Easy

4. How is the effective access time computed for a demand-paged memory system?

Ans: In order to compute the effective access time, it is necessary to know the average memory access time of the system, the probability of a page fault, and the time necessary to service a page fault. The effective access time can then be computed using the formula:
effective access time = (1 – probability of page fault) * memory access time + probability of page fault * page fault time.
Feedback: 10.2.3
Difficulty: Easy

5. Explain how copy-on-write operates.

Ans: Copy-on-write (COW) initially allows a parent and child process to share the same pages. As long as either process is only reading—and not modifying—the shared pages, both processes can share the same pages, thus increasing system efficiency. However, as soon as either process modifies a shared page, a copy of that shared page is created, thus providing each process with its own private page. For example, assume an integer X whose value is 5 is in a shared page marked as COW. The parent process then proceeds to modify X, changing its value to 10. Since this page is marked as COW, a copy of the page is created for the parent process, which changes the value of X to 10. The value of X remains at 5 for the child process.
Feedback: 10.3
Difficulty: Medium

6. Explain the usefulness of a modify bit.

Ans: A modify bit is associated with each page frame. If a frame is modified (i.e. written), the modify bit is then set. The modify bit is useful when a page is selected for replacement. If the bit is not set (the page was not modified), the page does not need to be written to disk. If the modify bit is set, the page needs to be written to disk when selected for replacement.
Feedback: 10.4.1
Difficulty: Easy

7. How does the second-chance algorithm for page replacement differ from the FIFO page replacement algorithm?

Ans: The second-chance algorithm is based on the FIFO replacement algorithm and even degenerates to FIFO in its worst-case scenario. In the second-chance algorithm, a FIFO replacement is implemented along with a reference bit. If the reference bit is set, then it is cleared, the page's arrival time is set to the current time, and the program moves along in a similar fashion through the pages until a page with a cleared reference bit is found and subsequently replaced.
Feedback: 10.4.5
Difficulty: Easy

8. Explain the distinction between global allocation versus local allocation.

Ans: When a process incurs a page fault, it must be allocated a new frame for bringing the faulting page into memory. The two general strategies for allocating a new frame are global and local allocation policies. In a global allocation scheme, a frame is allocated from any process in the system. Thus, if process A incurs a page fault, it may be allocated a page from process B. The page that is selected from process B may be based upon any of the page replacement algorithms such as LRU. Alternatively, a local allocation policy dictates that when a process incurs a page fault, it must select one of its own pages for replacement when allocating a new page.
Feedback: 10.5.3
Difficulty: Medium

9. Why doesn't a local replacement algorithm solve the problem of thrashing entirely?

Ans: With local replacement, if one process starts thrashing, it cannot steal frames from another process and cause the latter to thrash as well. However, if processes are thrashing, they will be in the queue for the paging device most of the time. The average service time for a page fault will increase because of the longer average queue for the paging device. Thus, the effective access time will increase, even for a process that is not thrashing.
Feedback: 10.6
Difficulty: Medium

10. What are the benefits of using slab allocation to allocate kernel memory?

Ans: The slab allocator provides two main benefits. First, no memory is wasted due to fragmentation. When the kernel requests memory for an object, the slab allocator returns the exact amount of memory required to represent the object. Second, memory requests can be satisfied quickly. Objects are created in advance and can be quickly allocated. Also, released objects are returned to the cache and marked as free, thus making them immediately available for subsequent requests.
Feedback: 10.8.2
Difficulty: Medium

11. Explain the concept behind prepaging.

Ans: Paging schemes, such as pure demand paging, result in large amounts of initial page faults as the process is started. Prepaging is an attempt to prevent this high level of initial paging by bringing into memory, at one time, all of the pages that will be needed by the process.
Feedback: 10.9.1
Difficulty: Medium

12. Discuss two strategies for increasing TLB reach.

Ans: TLB reach refers to the amount of memory accessible from the TLB and is the page size multiplied by the number of entries in the TLB. Two possible approaches for increasing TLB reach are (1) increasing the number of entries in the TLB, and (2) increasing the page size. Increasing the number of entries in the TLB is a costly strategy as the TLB consists of associative memory, which is both costly and power hungry. For example, by doubling the number of entries in the TLB, the TLB reach is doubled. However, increasing the page size (or providing multiple page sizes) allows system designers to maintain the size of the TLB, and yet significantly increase the TLB reach. For this reason, recent trends have moved towards increasing page sizes for increasing TLB reach.
Feedback: 10.9.3
Difficulty: Medium

13. How are lock bits useful in I/O requests?

Ans: A lock bit is associated with every frame. If a frame is locked, it cannot be selected for replacement. To write a block on tape, we lock into memory the pages containing the block. The system then continues as usual with other processes if the I/O request is in a queue for that I/O device. This avoids the replacement of the pages for other processes and the possible unavailability of those pages when the I/O request advances to the head of the device queue. When the I/O is complete, the pages are unlocked.
Feedback: 10.9.6
Difficulty: Medium

14. Explain how working set model works.

Ans: The working set model uses an integer parameter Δ. The set of pages in the most recent Δ references is the working set of a process. A process is allowed to run only if its entire working set can be accommodated in memory. Otherwise, the process is swapped out. If there are enough extra frames in memory, another process can be initiated.
Feedback: 10.6.2
Difficulty: EASY

True/False Questions

1. A page fault must be preceded by a TLB miss.

Ans: True
Feedback: 10.2.1
Difficulty: Easy

2. The instruction that causes a page fault needs to be re-executed after the fault has been handled.

Ans: True
Feedback: 10.2.1
Difficulty: Medium

3. Stack algorithms can never exhibit Belady's anomaly.

Ans: True
Feedback: 10.4
Difficulty: Medium

4. Some operating systems keep a pool of free frames so that the frequency of page faults is lowered.

Ans: False
Feedback: 10.4.7
Difficulty: Medium

5. Some operating systems provide raw disk, so that special applications can bypass file system when accessing secondary storage.

Ans: True
Feedback: 10.4.8
Difficulty: Difficult

6. A reaper starts reclaiming pages as soon as the number of free frames falls below the maximum threshold.

Ans: False
Feedback: 10.5.3
Difficulty: Medium

7. The current best practice to avoid thrashing is to include enough physical memory.

Ans: True
Feedback: 10.6.4
Difficulty: Easy

8. The buddy system for allocating kernel memory is very likely to cause fragmentation within the allocated segments.

Ans: True
Feedback: 10.8.1
Difficulty: Easy

9. In Linux, a slab may only be either full or empty.

Ans: False
Feedback: 10.8.2
Difficulty: Medium

10. Prepaging an executable program is much easier than prepaging a text file.

Ans: False
Feedback: 10.9.1
Difficulty: Easy

11. Solaris uses both a local and global page replacement policy.

Ans: False
Feedback: 10.10.3
Difficulty: Medium

12. Windows uses both a local and global page replacement policy.

Ans: False
Feedback: 10.10.2
Difficulty: Easy