No calculators. 110 points graded out of 100 points.

1. The following diagram has two of the locality labels misplaced. Circle the two that are misplaced. (6 pts.)

2. Identify each of the two address translation approaches shown below. Note that we call “system address” a virtual address, “primary address” a physical address, and “word” an offset. We also have different names for “block” and “lookup table”. Nevertheless, the actions shown in each diagram will allow you to distinguish between the two approaches. (4 pts.)

   a) _____ paging _______  b) _____ segmentation _______

Static relocation/Dynamic relocation. Circle one of S or D. (2 pts. each)
3. SR / DR Paging on modern computers.
4. SR / DR Roll-out/roll-in on IBM computers in 1960s.
5. SR / DR Historical program load-and-execute approach of 1940s and early 1950s.
6. SR / DR Swapping on CDC computers in 1960s with one set of relocation registers (i.e., single segment approach).

   4 and 5 are SR; 3 and 6 are DR

Static allocation/Dynamic allocation/Partial allocation. Circle one of S, D, or P, as applies. (3 pts. each)
7. S / D / P Paging on modern computers.
10. S / D / P Swapping on CDC computers in 1960s with one set of relocation registers (i.e., single segment).

   9 is S; 8 and 10 are D; 7 is P
Paging/Segmentation. Circle one or both of P or S, as applies. (2 pts. each)
11.  P  /  S   Use variable-length allocation.
12.  P  /  S   Can have internal fragmentation.
13.  P  /  S   Performs a bounds check on the address.
15.  P  /  S   Process control block holds the base address of the mapping table.
16.  P  /  S   Requires a special register in the CPU to hold the base address of the mapping table.

12 is P only; 11 and 13 are S only; 14, 15, and 16 are both P and S

True/False. Circle either T or F. (2 pts. each)
17.  T  /  F   A page or segment fault will always be immediately preceded by a TLB miss.
18.  T  /  F   Dynamic linking requires the same hardware support as dynamic relocation.
19.  T  /  F   Copy-on-write allows parent and child processes to share a page until one of them writes to it.
20.  T  /  F   Before a program is run, a paging system creates the process memory image in physical memory.
21.  T  /  F   A page table entry in the page table requires a presence bit, as does the page table entry in the TLB.
22.  T  /  F   A working set is the set of pages referenced by a program since the time the program first started to execute.

17 and 19 are true; 18, 20, 21, and 22 are false

23. Label the following steps required in processing a page fault in their proper sequential order, 1-4. (4 pts.)
___ OS eventually dispatches to process 1. Load instruction in process 1 is retried and succeeds.
___ I/O completion interrupt for the page-in; OS sets the presence bit in the PTE; and, OS changes process 1 back to ready.
___ Load instruction in process 1 has a page fault. CPU saves information about the fault and context switches to an OS fault handler. OS checks the PTE to see if requested access is permitted. Since access is allowed, OS does not terminate process 1.
___ OS finds a free page frame or selects which page frame should be replaced, and schedules an I/O operation to bring in the missing page from disk to the selected page frame. OS suspends process 1. OS dispatches to another ready process (e.g., process 2).

24. Consider a 10-page virtual memory with virtual pages numbered from 0 to 9, 10-word pages with words numbered from 0 to 9, a 4-page physical memory with page frames numbered from 0 to 3.

VPN P R W X PFN
+-------+-------+-------+-------+-------+
0|0|1|0|0|0 | P - presence bit  0 = not present, 1 = present
1|1|1|0|1|2 | R - read permission 0 = not allowed, 1 = allowed
2|1|1|1|0|1 | W - write permission 0 = not allowed, 1 = allowed
3|0|1|1|0|0 | X - execute permission 0 = not allowed, 1 = allowed
4|1|1|0|0|0 |
5|0|1|0|0|0 | PFN - page frame number
...  ...  ...

Starting each time from this page table, what is the result of these virtual address accesses - give either the physical address or the system action (e.g., page fault or protection violation). (3 pts. each)

a. inst. fetch 15 fetch from physical address 25
b. read 32 page fault (reads are permitted but page is not present in main memory)
c. read 45 read from physical address 05
d. write 47 protection error (page is present but writes are not permitted)
25. What can be done to prevent having to flush the TLB on each process switch? (3 pts.)
   prepend process id bits to each virtual address to disambiguate the virtual addresses
   (requires a process id register in the CPU, which will be loaded by the dispatcher on a process switch)

26. What is memory “compaction” (or “shuffling”)? (3 pts.)
   move all allocated blocks of memory to one end, thus creating a single free block at the other end

27. What is memory “thrashing”? (3 pts.)
   system is consumed with paging traffic, with little to no useful work being done

28. How does PFF decide to steal pages from one process to give to another? (5 pts.)
   PFF divides the processes into three sets:
   1) those with page fault frequencies above a higher threshold
   2) those with page fault frequencies between the thresholds
   3) those with page fault frequencies below a lower threshold
   When there are processes in set 1, PFF attempts to steal pages from processes in set 3. If there are no processes in set 3, PFF causes the OS to reduce the number of processes competing for main memory.

29. Consider a byte-addressable computer system with a 32-bit virtual address, a page size of 4096 bytes, and a physical memory of 16 MB. (3 pts. each)
   a) How many bits are in the physical address? \( \log_2(16M) = 24 \), so 24 bits in physical address
   b) How many bits are in the page frame number? \( \log_2(4096) = 12 \), so 12 bits in offset; 24 bits in physical address – 12 bits in offset = 12 bits in PFN
   c) How many bits are in the virtual page number? 32 bits in virtual address – 12 bits in offset = 20 bits in VPN
   d) How many entries are in an inverted page table? IPT size is based on physical memory size, not virtual memory size; thus one entry per PFN = \( 2^{12} = 4096 \)

30. The diagram below represents FIFO replacement with a victim buffer. On a page fault, a page is mapped to main memory and placed at the head of the mapped page list. Main memory has six page frames, but only four mapped pages can exist at one time. (The L-to-R links represent next-node pointers, and the R-to-L links represent previous-node pointers.)

   ![Diagram](image)
   a) Draw the FIFO with victim buffer data structure that results after a page reference to page 2. Is a page-in from disk required? (4 pts.)

   The reference to page 2 is a page fault, but the OS can do a fast reclaim of page 2 from the victim buffer. Thus, no page-in from disk is required.
   (Page 2 is removed from the victim buffer and placed at the head of the list. Pages 3, 0, 7, and 1 are shifted right in the list, with page 1 being unmapped as it enters the victim buffer. Page 4 remains as the replacement victim.)
b) Starting from the original diagram above (that is, ignore part (a)), draw the FIFO with victim buffer data structure that results after a page reference to page 5. Is a page-in from disk required? (4 pts.)

The reference to page 5 is a page fault, and the OS must read page 5 from disk. (Page 5 replaces the replacement victim page 4, and page 5 is then placed at the head of the list. Pages 3, 0, 7, 1, and 2 are shifted right in the list, with page 1 being unmapped as it enters the victim buffer. Page 2 is the new replacement victim.)

31. The diagram below represents FIFO second chance replacement. Main memory has four page frames, and “ref” means the reference bit.

a) Draw the FIFO second chance data structure that results after a page reference to page 4. (3 pts.)

The reference to page 4 is a page hit. The reference bit for page 4 is set. The victim pointer does not move.

b) Starting from the original diagram above (that is, ignore part (a)), draw the FIFO second chance data structure that results after a page reference to page 2. (3 pts.)

(The reference to page 2 is a page fault. The OS examines the replacement victim page 8, but page 8’s reference bit is set. The OS resets page 8’s reference bit and advances the victim pointer, thus giving page 8 a second chance. The OS examines the replacement victim page 4. Page 4’s reference bit is clear, so the OS replaces page 4 with page 2. The OS advances the victim pointer so that page 2 will be allowed to stay in memory until the victim pointer makes at least one more FIFO cycle, and page 2’s reference bit is left clear so that a subsequent reference can be recorded.)