COSC 450 Operating System Mini-Test #3

10/29/2024

Name:_____.

1. (2 pt.)
    a. A system uses a bitmap to track free memory space. Currently, the system maintains a bitmap of size 8KB, with each unit being 2KB. What is the size of the RAM?

Sol)

- Since Bit map size is 8KB = $8 \times 2^{10} \times 8$ (bit), there are 64×2^{10} units.
- Total RAM size = # of unit $\times$ size of unit = $(64 \times 2^{10}) \times (2 \times 2^{10}) = 128 \times 2^{20}$Byte = 128 MB

    b. In the same system, a new operating system has been installed that utilizes a free-list to manage free memory. Assume the memory currently consists of an alternating sequence of segments and holes, each measuring 64 KB. Additionally, assume that each node in the linked list requires a 32-bit memory address, a 16-bit length field, and a 16-bit field for the next node. How many bytes of storage are needed for the free list?

Sol)

- The linked list: number of node for linked list= 128 MB /64KB = $2^{27}/2^{16}$ or 2^{11} nodes.
- size of each node = 32+16+16 = 64 bit = 8 byte = 2^3 bytes
- Total size of linked list = number of node $\times$ size of a node = $2^{11} \times 2^3$ bytes = **2^{14} bytes**.

2. (1 pt.) Consider a swapping system in which memory consists of the following hole sized in memory order: 21KB, 20KB, 4KB, 18KB, 15KB, 14KB, 25KB, 23KB and 35KB.

Which hole is taken for successive segment request of 9K, 10KB, 15KB and 18KB for first fit? Now repeat the question for best fit, worst fit and next fit.

◊ First Fit 9(21) -10(12) -15(20)- 18(18)

◊ Best Fit 9(14) – 10(15)-15(18)-18(20)

◊ Worst Fit 9 (35) – 10(26) -15 (25) – 18 (23)

◊ Next Fit 9(21)-10(20)-15(18) -18(25) or
9(21) -10(12)- 15(20) -18(18)

3. (2 pt.) How many page faults occur for following reference string with three page frame for each of following page replacement algorithm? Lets assume page frames are initially empty. (Show detailed steps)

5, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 5, 0, 1

a) FIFO : 15 page fault

5,	0,	1,	2,	0,	3,	0,	4,	2,	3,	0,	3,	2,	1,	2,	0,	1,	5,	0,	1
5	0	1	2		3	0	4	2	3	0			1	2			5	0	1
	5	0	1		2	3	0	4	2	3			0	1			2	5	0
		5	0		1	2	3	0	4	2			3	0			1	2	5

b) LRU: 12 page fault

5,	0,	1,	2,	0,	3,	0,	4,	2,	3,	0,	3,	2,	1,	2,	0,	1,	5,	0,	1
5	0	1	2		2		4	4	4	0			1		1		1		
	5	0	1		3		3	2	2	2			2		2		5		
		5	0		0		0	0	3	3			3		0		0		

c) Optimal: 9 page fault

5,	0,	1,	2,	0,	3,	0,	4,	2,	3,	0,	3,	2,	1,	2,	0,	1,	5,	0,	1
5	0	1	1		3		3			3			1				1		
	5	0	0		0		4			0			0				0		
		5	2		2		2			2			2				5		

4. (2 pt.)
   a. A computer system generates a 32-bit virtual address for a process, with each entry in the page table requiring 64 bits. The system has a page table size of 8 MB. Based on this information, what is the size of each page in KB?

   Sol) # of page entries = size of page table / entry size = 8MB / (64/8) = 2^{20} entries
   Page size = possible virtual space / # entries = 2^{32} /2^{20} = 4×2^{10} = 4 KB

   b. In same system, 21 bits are used for saving page frame number in page table. What is the current RAM size?
   Sol) # of page frame = 2^{21} page frames
   Size of RAM = size of page frame $\times$ number of page frames
   = 4KB $\times$ 2^{21} = $4 \times 2^{10} \times 2^{21} = 8 \times 2^{30}$ = 8 GB

5. (1 pt.) An operating system might support three types of scheduler. Briefly discuss three types of schedulers.

- Long-Term Scheduler – Selects a process from the pool of job and load into memory for execution
- Short-term scheduler – selects a process from the ready queue and allocates the CPU.
- Memory Scheduler – schedule which process is in memory and in the secondary memory of swap area.

6. (1 pt.) Why maintaining page table in memory can result in slower memory access time?

Sol) To access an instruction located in virtual address I, OS need two memory access times.
- Access a memory to get page frame number from page table in the memory
- Now calculate physical address by combining page frame number + offset

7. (1 pt.) Page size is one of most important design issues in the operating system. We can mathematically analyze page size based on following assumptions:
   - S: average size of process (byte)
   - P: the size of page (byte)
   - E: Each page entry requires (byte)
   - 50% of memory in the last page of the process is wasted due to internal fragmentation

   a. Define the total overhead function based on page size P.

   $$Total\ Overhead(P) = Average\ page\ table ¿ wasted\ memory \in th\ last\ page\ of\ process = \frac{S}{P} \times E + \frac{P}{2}$$

   b. Find the optimal page size formula based on the total overhead (by minimize the total overhead)

   $$Overhead'(P) = -\frac{SE}{P^2} + \frac{1}{2} = 0$$

   $$P = \sqrt{2SE} : optimal\ \text{page size}$$