CS 370: Operating Systems

Colorado State University FALL 2016

INSTRUCTOR: Yashwant K Malaiya

HW2

DUE DATE: Thursday Dec 1 2016 5 PM. Submit MS Word or PDF file using Canvas. For all questions, you must show how the answer was obtained by giving the main steps in the procedure.

Problem 1 (13 points) Recall the various deadlock detection and prevention algorithms we've discussed in this course, and consider the following snapshot of a system with five processes (P1, P2, P3, P4, P5) and four resources (R1, R2, R3, R4).

There are no current outstanding queued unsatisfied requests.

Currently Available Resources

R1	**R2**	**R3**	**R4**
2	1	2	0

	Current Allocation				**Max Need**				**Still Needs**			
Process	**R1**	**R2**	**R3**	**R4**	**R1**	**R2**	**R3**	**R4**	**R1**	**R2**	**R3**	**R4**
P1	0	0	1	2	0	0	3	2	0	0	2	0
P2	2	0	0	0	2	7	5	0	0	7	5	0
P3	0	0	3	4	6	6	5	6	6	6	2	2
P4	2	3	5	4	4	3	5	6	2	0	0	2
P5	0	3	3	2	0	6	5	2	0	3	2	0

i) Is this system currently deadlocked, or can any process become deadlocked? Why or why not? If not deadlocked, give an execution order.

No, because P1 can immediately get satisfied by using free resources and released then so on as other processes.

P1 —> P4 —> P5 —> P2 —> P3

ii) If a request from a process P1 arrives for (0, 4, 2, 0), can the request be immediately granted? Why or why not? If yes, show an execution order.

No, there is not enough resources available. It is not valid.

iii) If a request from a process P2 arrives for (0, 1, 2, 0), can the request be immediately granted? Why or why not? If yes, show an execution order.

No, the request can be granted but it's unsafe and has not execution order because there is not enough free resources for any sequences of processes to complete under such circumstance.

Problem 2 (12) A system has these successive page accesses: 1 2 3 4 2 3 4 1 2 1 1 3 1 4 and that there are three frames within the system. Demonstrate the use of the **FIFO** replacement algorithm. Use the table below to the successive contents of the frames. When there is no page faults, leave the column blank.

Answer:

	1	2	3	4	2	3	4	1	2	1	1	3	1	4	
F0	1	1	1	4	4	4	4	4	4	4	4	3	3	3	
F1		2	2	2	2	2	2	1	1	1	1	1	1	4	
F2			3	3	3	3	3	3	2	2	2	2	2	2	

Problem 3 (12) A system has these successive page accesses: 1 2 3 4 2 3 4 1 2 1 1 3 1 4 and there with three page frames. Demonstrate the use of the **LRU** replacement algorithm Use the table below to the successive contents of the frames. When there is no page faults, leave the column blank.

CS 370: Operating Systems

Colorado State University FALL 2016

INSTRUCTOR: Yashwant K Malaiya

Answer:

	1	2	3	4	2	3	4	1	2	1	1	3	1	4	
F0	1	1	1	4	4	4	4	4	4	4	4	3	3	3	
F1		2	2	2	2	2	2	1	1	1	1	1	1	1	
F2			3	3	3	3	3	3	2	2	2	2	2	4	

Problem 4 (13 points) A system has a demand-paged memory. Accessing the page table takes negligible time. It takes 8 milliseconds to service a page fault if an empty page is available or the replaced page is not modified, and 20 milliseconds if the replaced page is modified. Memory access time is 100 nanoseconds. Assume that the page to be replaced is modified 70 percent of the time. What is the maximum acceptable page-fault rate for an effective access time of no more than 200 nanoseconds?

Work:

Let p be the page fault rate (the probability that a memory access results in a page fault).

effective access time = 200 ns = 100(1-p) + p(0.3*8+0.7*20)*10^6 = 200

Thus p = 100 / 16399900 = 0.000006098

Answer: p = 6.098 * 10 ^ -6

Problem 5 (13) The page table below is for a system with 16-bit virtual and physical addresses and with 4,096-byte pages. The reference bit is set to 1 when the page has been referenced. Periodically, a thread zeroes out all values of the reference bit. A dash for a page frame indicates the page is not in memory. The page-replacement algorithm is localized LRU, and all numbers are provided in decimal.

CS 370: Operating Systems

Colorado State University FALL 2016

INSTRUCTOR: Yashwant K Malaiya

Page	PageFrame	Reference Bit
0	9	0
1	1	0
2	14	0
3	10	0
4	–	0
5	13	0
6	8	0
7	15	0
8	–	0
9	0	0
10	5	0
11	4	0
12	–	0
13	–	0
14	3	0
15	2	0

Answer the following questions.

A. Convert the following virtual addresses (in hexadecimal) to the equivalent physical addresses(hex).

Virtual Address	Physical Address
0xE12C	0x312C
0x3A9D	0xAA9D
0xA9D9	0x59D9
0x7001	0xF001
0xACA1	0x5CA1

CS 370: Operating Systems

Colorado State University FALL 2016

INSTRUCTOR: Yashwant K Malaiya

B. Using the above addresses as a guide, provide an example of a logical address (in hexadecimal) that results in a page fault.

Answer:

0x4444

C. From what set of page frames will the LRU page-replacement algorithm choose in resolving a page fault?

Answer:

{9, 1, 14, 13, 8, 0, 4}

Problem 6 (12) A demand-paging system with a paging disk that has an average access and transfer time of **20 milliseconds**. Addresses are translated through a page table in main memory, with an access time of **1 microsecond** per memory access. Thus, each memory reference through the page table takes **two accesses**. To improve this time, we have added an associative memory that reduces access time to one memory reference, if the page-table entry is in the associative memory. Assume that **80 percent of the accesses** are in the associative memory and that, of those remaining, 10 percent (or **2 percent of the total**) cause page faults. What is the effective memory access time?

Answer: 401.2 microsecond

effective memory access time = 0.8 * 1 mic + 0.18 * 2 mic + 0.02 * (20 milli + 2 mic)
= 401.2 mic

Problem 7 (13) Consider the following page reference string:
7, 2, 3, 1, 2, 5, 3, 4, 6, 7, 7, 1, 0, 5, 4, 6, 2, 3, 0 , 1.
Assuming demand paging with three frames, how many page faults would occur for the following replacement algorithms? Also trace the sequences.

- LRU replacement
- FIFO replacement
- Optimal replacement

CS 370: Operating Systems

Colorado State University FALL 2016

INSTRUCTOR: Yashwant K Malaiya

	7	2	3	1	2	5	3	4	6	7	7	1	0	5	4	6	2	3	0	1	
F0	7	7	7	1	1	1	3	3	3	7	7	7	7	5	5	5	2	2	2	1	
F1		2	2	2	2	2	2	4	4	4	4	1	1	1	4	4	4	3	3	3	
F2			3	3	3	5	5	5	6	6	6	6	0	0	0	6	6	6	0	0	
	1	2	3	4	4	5	6	7	8	9	9	10	11	12	13	14	15	16	17	18	

LRU: Answer: 18

	7	2	3	1	2	5	3	4	6	7	7	1	0	5	4	6	2	3	0	1	
F0	7	7	7	1	1	1	1	1	6	6	6	6	0	0	0	6	6	6	0	0	
F1		2	2	2	2	5	5	5	5	7	7	7	7	5	5	5	2	2	2	1	
F2			3	3	3	3	3	4	4	4	4	1	1	1	4	4	4	3	3	3	
	1	2	3	4	4	5	5	6	7	8	8	9	10	11	12	13	14	15	16	17	

FIFO: Answer: 17

	7	2	3	1	2	5	3	4	6	7	7	1	0	5	4	6	2	3	0	1	
F0	7	7	7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
F1		2	2	2	2	5	5	5	5	5	5	5	5	5	4	6	2	3	3	3	
F2			3	3	3	3	3	4	6	7	7	7	0	0	0	0	0	0	0	0	
	1	2	3	4	4	5	5	6	7	8	8	8	9	9	10	11	12	13	13	13	

CS 370: Operating Systems

Colorado State University FALL 2016

INSTRUCTOR: Yashwant K Malaiya

Optimal: Answer: 13

Problem 8 (12 points) A disk queue is holding requests to the these cylinders in the order: 116, 22, 3, 11, 75, 185, 100, 87. Using the SCAN scheduling algorithm, what is the order in which the requests are serviced, assuming the disk head is at cylinder 88 and moving upward through the cylinders?

Use the graph below for a diagram.

Answer: 88 - 100 - 116 - 185 - 87 - 75 - 22 - 11 - 3

88	88	88	88	88	88	88	88	88		
	100	100	100	100	100	100	100	100		
		116	116	116	116	116	116	116		
			185	185	185	185	185	185		
				87	87	87	87	87		
					75	75	75	75		
						22	22	22		
							11	11		
								3		