CS 4310 Operating Systems
Exam 2
Max: 200 points
(12/12/2024)

Name: Loc Nguyen

Read these instructions before proceeding.

- Closed book. Closed notes. You can use calculator.
- You have 110 minutes to complete this exam.
- *Important Notes:*
  - *Box your answers.*
  - *No questions will be answered during the exam period about the exam questions. Write down your assumptions and answer the best that you can.*
  - *Just in case you have trouble of submitting your exam here @Canvas, alternative way is to submit your completed exam to Prof. Young by emailing*
    gsyoung@cpp.edu
- You need to submit your completed exam paper in ***one PDF file***.
  Two popular ways that students work on the exam are:
  - (1) Print out the exam paper. Write your answers on the exam paper. Scan your completed exam papers or take photos of them. Then turn **in *one PDF file*** here @ *Canvas*.
  - (2) Read the exam from the computer screen and answer questions on your own white papers (number your answers). Scan your exam answers or take photos of them. Then turn in ***one PDF file*** here @ *Canvas*.
- Answer the problems on the blank spaces provided for each problem.

Q.#1 (40)	Q.#2 (40)	Q.#3 (40)	Q.#4 (40)	Q.#5 (40)	Total (200)

1. (40 points) Fill in the blanks & short answer

(a) (4 pts) Files whose bytes or records can be read in any order are called **Random** access files.

(b) (4 pts) When the computer is booted, the BIOS reads in and executes Section 0 of the disk, called **Master Boot Record (MBR)**.

(c) (4 pts) A compute with a 32-bit address uses a three-level page table. Virtual addresses are split into a 6-bit top-level page table field, a 5-bit second-level page table field, a 5-bit third-level page table field, and an offset.

How many pages are there in the address space? **2^{16} number of pages**

(d) (4 pts) In theory, we can build secure systems as long as we keep the computer systems simple. However, as we introduce more **features**, more complexity arises, thus compromising the ability to develop a secure system.

(e) (4 pts) MD5 is a cryptographic hash function that produces a 16-byte result. Given a result (output), the practical infeasibility of brute force guessing in the worst-case scenario requires that we must guess an input **2^{128}** number of times in order to find a match.

(f) (4 pts) Stack algorithm, such as LRU, does not suffer from **Belady's Anomaly**.

(g) (h) (i) (j) (16 pts)

A deadlock situation can arise if and only if the following four conditions hold simultaneously in a system. (Coffman et al.)

1. **Mutual Exclusion**
2. **Hold and Wait**
3. **No preemption**
4. **Circular wait**

2. (40 points) Consider a swapping system in which memory consists of the following holes sizes in memory order: 11 KB, 13 KB, 10 KB, 12 KB, 14 KB, 15 KB, and 26 KB. Which hole is taken for successive segment requests of

12 KB,
13 KB,
10 KB,
11 KB,

(a) for first-fit?
13KB hole – 12KB
14KB hole – 13KB
11KB hole – 10KB
12KB hole – 11KB

(b) for worst-fit?

26KB hole (14KB remain) – 12KB

15KB hole (2KB remain) – 13 KB

14KB hole (4KB remain) – 10 KB

13KB hole (2KB remain) – 11KB

(c) for best-fit?

12KB hole – 12KB

13KB hole – 13KB

10KB hole – 10KB

11KB hole – 11KB

(d) for next-fit?

13KB hole (1KB remain) – 12 KB

14KB hole (1KB remain) – 13KB

15KB hole (5KB remain) – 10KB

26KB hole (15KB remain) – 11KB

3. (40 points)

(a) If ***FIFO*** page replacement is used with four page frames and eight pages, how many page faults will occur with the reference string 01234016457365 if four frames are initially empty? *Show all your steps.*

There is 11 page faults using FIFO

0	1	2	3	4	0	1	6	4	5	7	3	6	5
0	0	0	0	4	4	4	4	4	5	5	5	5	5
	1	1	1	1	0	0	0	0	0	7	7	7	7
		2	2	2	2	1	1	1	1	1	3	3	3
			3	3	3	3	6	6	6	6	6	6	6

(b) Repeat the problem in part (a) for ***LRU***. *Show all your steps*

There is 12 page faults using LRU

0	1	2	3	4	0	1	6	4	5	7	3	6	5
0	0	0	0	4	4	4	4	4	4	4	4	6	6
	1	1	1	1	0	0	0	0	5	5	5	5	5
		2	2	2	2	1	1	1	1	7	7	7	7
			3	3	3	3	6	6	6	6	3	3	3

4. (40 points)

	Has	Max
A	**1**	**3**
B	**0**	**1**
C	**2**	**6**
D	**2**	**7**
E	**1**	**3**
Free: 2		

Take a careful look at the above. Use the Banker's Algorithm for a Single Resource to determine if each of the following requests leads to a safe state or an unsafe state.

(a) If ***C*** asks for one more unit, does this lead to a safe state or an unsafe state? Justify your answer by showing all your steps.

If C asks for one more unit, it will still be in a safe state because process B can still be completed. Refer to the table below for justification
Current table state which has two free units and process A or E can complete

	Has	Max	Need
A	1	3	2
B	0	1	1
C	2	6	4
D	2	7	5
E	1	3	2

Table state after C asks for one more unit and only one free unit remaining. Process B can still complete so machine is still in a safe state

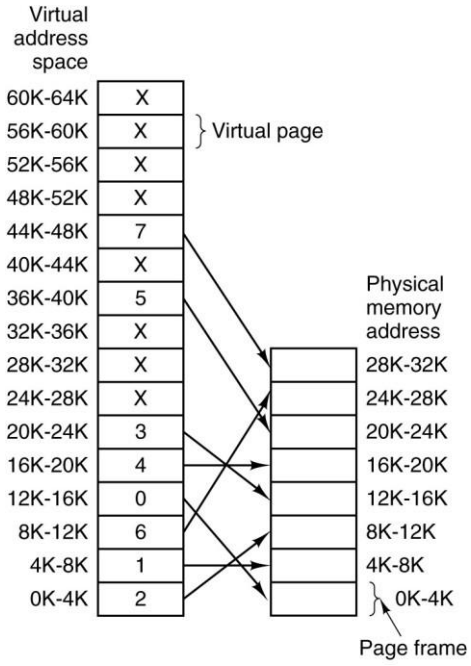
	Has	Max	Need
A	1	3	2
B	0	1	1
C	3	6	3
D	2	7	5
E	1	3	2

(b) If ***E*** asks for one more unit (instead of C), does this lead to a safe state or an unsafe state? Justify your answer by showing all your steps
If E asks for one more unit instead of C, it will still be in a safe state. Refer to the table below after E asks for one more unit

	Has	Max	Need
A	1	3	2
B	0	1	1
C	2	6	4
D	2	7	5
E	2	3	1

Process B and E can still use the remaining unit to finish so machine is in safe state

5. (40 points) A computer has 16-bit virtual addresses and 4-KB pages. It has 32 KB physical memory. A snap shot of the mapping from pages to page frames is as follows.



Calculate the physical address for each of following virtual addresses:

a) virtual address 21

Virtual address 21 belongs to virtual page 0 and virtual page 0 is mapped to physical frame 2. Therefore, the physical address would be = 8192 + (21-0) = 9013

b) virtual address 4097

Virtual address 4097 belongs to virtual page 1 and virtual page 1 is mapped to physical frame 1. Therefore, the physical address would be = 4096 + (4097 – 4096) = 4097

c) virtual address 13002

Virtual address 13002 belongs to virtual page 3 and virtual page 3 is mapped to physical frame 0. Therefore, the physical address would be = 0 + (13002 – 12288) = 714

d) virtual address 20003

Virtual address 20003 belongs to virtual page 4 and virtual page 4 is mapped to physical frame 4. Therefore, the physical address would be = 16384 + (20003 – 16384) = 20003