VIII. Security (10 points total, FOR 541 STUDENTS ONLY)

Explain your solution to the following question. You do not need to calculate the keys using any encryption algorithm, just state how the message is encrypted and decrypted for the task.

Alice (A), Bob (B), and Charlene (C) together want to send a message to Doug (D). They want this message to be private, so that only Doug can read it. They also want Doug to be able to show that the message came from all three of Alice, Bob, and Charlene. They decide to use public-key encryption to do this. Each of the three persons have their own private keys. The public keys for each person are reliably available to any of the other people.

Alice, Bob, and Charlene write the message together. Describe how they would use the public-key encryption to prepare the message for Doug. Also describe what Doug does to read and verify the message. Make sure that you describe how Doug knows which keys to try.

1. Which of the following is not a condition necessary for deadlock:
A. Circular wait
B. Starvation
C. Mutual exclusion
D. Hold and wait

2. Assuming process execution times are known in advance, for which of the following algorithms can the maximum wait time for a given process be computed at the time the job is submitted?
A. Shortest job first
B. First-come, first-served
C. Priority-based
D. Random scheduling

3. Which of the following always achieves the optimal performance of memory allocation?
A. Best fit
B. First fit
C. Worst fit
D. None of above

4. The best description of the purpose of a TLB is:
A. to cache page translation information
B. to cache frequently used data
C. to hold variable values for a running process
D. to hold start and length of the page table

5. Which of the following is NOT usually stored in a two-level page table:
A. page offset
B. inner page number
C. outer page number
D. virtual page number

II. ***True/False (Give a one-sentence explanation to your answer. 1 point each for True/ False, 1 point each for explanation. 26 points total)***

1. A page frame is a portion of main memory.

2. RSA encryption can avoid man-in-middle attack.

3. External fragmentation can be prevented (almost completely) by frequent use of compaction, but the cost would be too high for most systems.

2. What information is used by the Least Recently Used page replacement policy?

3. In terms of memory allocation, what is a reference counter? Why is it needed?

4. Explain why, or why not, internal fragmentation can be a problem when using the best fit algorithm for memory allocation.

5. What would be the effect of a large number of page faults by a process on that process's page allocation on a nonpreemptive operating system?

IV. File management (9 points total)

Calculate the maximal size of a file if we have:

10 direct block address
one single indirect block address
one double indirect block address
one triple indirect block address
one quadrupple indirect block addresses

The size of a block is 512 bytes. The size of an address is 8 bytes.

a) What is the maximum size of a file? (6 points)

c. (514 - 4 points, 414 – 5 points) Does safety in the bankers' algorithm eliminate *all* possible kinds of deadlock? Why or why not?

d. (514 - 4 points, 414 – 5 points) Does lack of safety in the bankers' algorithm *guarantee* deadlock? Why or why not?

VI. Page replacement (10 points for 514 students, 15 points for 414 students)

Given four page frames and the memory reference string below, show what will be in each page frame after each reference using the following page replacement algorithms.

a) FIFO(514 - 3 points, 414 – 5 points)

	2	1	3	4	0	2	4	1	1	3	0	4	2	5	5	3
P0																
P1																
P2																
P3																

How many page faults?

b) LRU (514 - 3 points, 414 – 5 points)

	2	1	3	4	0	2	4	1	1	3	0	4	2	5	5	3
P0																
P1																
P2																
P3																

How many page faults?

4. Once a virtual memory page is loaded into main memory, it cannot be written to the disk.

5. Pages that are shared between two or more processes can never be swapped out to the disk.

6. Demand paging requires the programmer to take specific action to force the operating system to load a particular virtual memory page.

7. Prepaging is one possibility for the fetch policy in a virtual memory system.

8. To create a digital signature, a sender first applies a hash function to the original plaintext message.

9. Tabular noncontiguous file allocation stores pointers to file blocks contiguously in tables to reduce the number of length seeks required to access a particular record.

10. External fragmentation can occur in a paged virtual memory system and cannot occur in a segmentation virtual memory system.

11. The translation lookaside buffer (TLB) is a software data structure that supports the virtual memory address translation operation.

12. The resident part of a process in the main memory can be changed in response to actions by other processes.

13. Interrupt handler is not allowed to disable all interrupts, otherwise the process that was interrupted will never execute again and there will be a deadlock.

III. Short answers (2 points each, 10 points total)

1. What problems could occur if virtual memory pages are always allocated in groups of four?

c) Optimal (514 - 4 points, 414 – 5 points)

	2	1	3	4	0	2	4	1	1	3	0	4	2	5	5	3
P0																
P1																
P2																
P3																

How many page faults?

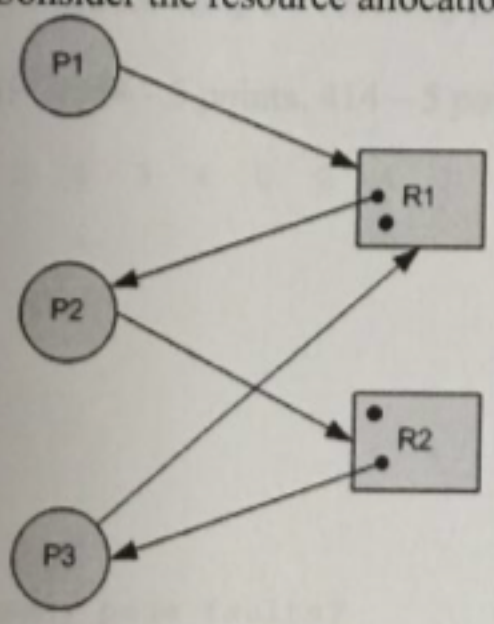
VII. Project description (10 points total)

Describe what data structures, software, and implementation methods you have used in your project #2 Simple Operating System.

b) If this file system has a file of 500MB, explain what block address will be used. (3 points)

V. *Deadlock prevention (15 points for 514 students, 20 points for 414 students)*

Consider the resource allocation diagram:



a. (514 - 3 points, 414 – 5 points) Is this situation deadlocked? Why or why not?

b. (514 - 4 points, 414 – 5 points) Is this situation "safe" according to bankers' algorithm (is there a way for *all* processes to complete)? Why or why not?