

Austin Brown

1a.

Internal Fragmentation – occurs when too much memory is given to a process. Thus, some of the memory is not used. The best way to deal with this is to assign as just enough memory for the process.

External Fragmentation – The memory space that is allocated to the process is not contiguous. This means the memory regions are not touching. This can be fixed or reduced by dynamically relocating the memory together.

1b.

- Increase in page size decreases the number of frames.
- The larger the page size, the fewer the TLB misses.
- The larger the page size, the more internal fragmentation you have.

1c.

The purpose of a working set algorithm is to guess what a program's locality is by looking at a working set. One of the most important considerations is page size. You could have an overlap of your pages are too large. You may never reach locality if they are too small. A working set is simply the most recent pages that are active. When a page is deactivated, it is dropped after a certain amount of time.

One way to implement this would be to first examine the local set size. The idea would be to allocate just enough space that the working set. We do not want to have extra frames. The working set should only have what it needs. We increase the allocated space to allow for the working set and decrease it when space is not needed.

2a.

$$2^5 = 32 \quad 2^6 = 64 \quad 2^7 = 128$$

$$32 + 2(64) + 2(128) = 416$$

$$4 \text{ bytes} * 416 = \boxed{1664 \text{ bytes}}$$

2b.

Most likely no. If you need various page tables of different sizes, then this would not be a good approach.

2c.

$$4(32 + 32(64) + 32(64)(128)) = \boxed{1056896 \text{ b}}$$

3a.

Round Robin					
Given Values		Work		Final Answer	
Process	Time	Process	Time	Process	Total Time
P1	13	P1	5	P1	54
P2	5	P2	5	P2	10
P3	23	P3	5	P3	81
P4	3	P4	3	P4	18
P5	31	P5	5	P5	92
P6	3	P6	3	P6	26
P7	14	P7	10	P7	68
		P1	10		
		P3	10		
		P5	10		
		P7	10		
		P1	13		
		P3	15		
		P5	15		
		P7	14		
		P3	20		
		P5	20		
		P3	23		
		P5	25		
		P5	30		
		P5	31		

First Come First Serve					
Given Values		Work		Final Answer	
Process	Time	Process	Time	Process	Total Time
P1	13	P1	13	P1	13
P2	5	P2	5	P2	18
P3	23	P3	23	P3	41
P4	3	P4	3	P4	44
P5	31	P5	31	P5	75
P6	3	P6	3	P6	78

[Type here]

P7	14	P7	14	P7	92
----	----	----	----	----	----

Shortest Job First					
Given Values		Work		Final Answer	
Process	Time	Process	Time	Process	Total Time
P1	13	P4	3	P1	24
P2	5	P6	3	P2	11
P3	23	P2	11	P3	61
P4	3	P1	13	P4	3
P5	31	P7	14	P5	92
P6	3	P3	23	P6	6
P7	14	P5	31	P7	38

4a.

$1 - 0.95 = .05 = \text{Miss Ratio}$

$0.95(10\text{ns} + 50\text{ns}) + 0.05(10\text{ns} + 50\text{ns}(3+1))$

$0.95(60\text{ns}) + 0.05(210\text{ns}) =$ 67.5 ns

4b.

$x(10+50) + (1-x)(10 + 50(3+1)) = 90\text{ns}$

$60x + (1-x)(210) = 90\text{ns}$

$60x + 210 - 210x = 90\text{ns}$

$-150x = -120$

$x = (-120/-150)$

$x =$.8 or 80% hit rate

4c.

$210 - 150x = 150\text{ns}$

$x = (-120/-150)$

$x =$ 0.4 or 40%

[Type here]

5a.

A critical section is an area that contains shared variables that multiple processes use. Because of this, they should only be accessed by one process at a time. Measures are taken to protect the critical section from multiple processes accessing at once.

5b.

Counting Semaphores – A semaphore allowing counts greater than 1. It is initially set to the max number of resources. Whenever the count is decremented, that means a process can use the resource. When the process is finished, the count is incremented.

Mutex Semaphores – Uses wait a signal to lock data. It is also known as a binary semaphore. This could be used if you want to lock a resource so that only one process can use it, and another process cannot.

Synchronization Semaphores – Used to synchronize data to ensure that other processes do not access that data at the wrong time. It could be used when you want all your processes to have an equal time using the shared memory. A process cannot have more time until every process has had a chance.

5c.

Yes. Even with a single core processor, you still have multiple processes running. Process one could be interrupted by process 2 at any time whether it is ready or not. Semaphores can be used a locking mechanism to keep that scenario from happening.