CS 405 Homework (part of Exam 3): Submit before class time 8 am - Tuesday 11/24.

Part A: (20)

Given the current state of the system:

Resource available with the system: 40

Process Name	Maximum Need	Allocated	Remaining Need
P1	100	60	40
P2	120	50	70
P3	80	50	30
P4	50	20	30

Evaluate each of the following request independently (on the above state). State if the request is safe and should be allocated. If so – show a sequence of allocation to complete all processes.

Process P3 makes a request for 30.

The System does have the available resources, and P3 would be able to finish afterward, so the resources would be allocated. The System would wait for P3 to finish because there are not enough resources to finish any other process. P3 finishes, releasing its 80 resource units totaling 90 available. P4 requests 30, which is available and still leaves enough to finish P1, so it is allocated leaving 60 available. P1 requests 40, leaving 20 available but P1 and P4 can still finish without further allocation, so it is allocated to P1. The program waits for P1 or P4 to finish. Either way when P1 or P4 finishes they will release enough resources to allocate to P2 requesting 70. Then the system waits for P2 to finish releasing all remaining resources and all processes are done.

P1 requests 40.

The System does have the available resources, and P1 would be able to finish afterward, so the resources would be allocated. The system would wait for P1 to finish, releasing 100 units. P2 requests 70, leaving 30 remaining units which is enough to finish P3 or P4, so it is allocated. P3 requests 30 units leaving 0 remaining, but P2 and P3 can still finish. The system waits for either P2 or P3 to finish, either of which would release enough resources to finish P4. P4 requests 30 units which are allocated because no other process is still running. Once P4 is done, it releases the remainder of resources and all processes are done.

P4 requests 20.

The System does have the available resources, and this would leave 20 units of resources left, which Is still enough for P4 to finish, so the resources would be allocated. P4 would request an additional 10 units, which would be allocated allowing P4 to finish. The system would wait for P4 to finish releasing 50 units, giving 60 available. P1 requests 40 units which would be allocated allowing P1 to finish. The System would wait for P1 to finish, releasing 100 units. P2 would request 70 of the 110 available. This would be allocated allowing P2 to finish and

leave enough for P3 to finish too. P3 would request 30 units allowing it to finish and all other processes are either done or already have enough resources, so it would be allocated. The System would wait for P2 and P3 to finish and which point all resources would be released and all processes would be done.

As these examples show, requesting a lot of resources at once can lock the system up for a time (not a deadlock though). This means that some processes may be requesting resources and just have to wait for another processes to finish and release their resources. If these processes instead requested smaller amounts of resources at a time, the processes could run at the same time and have to wait less. The requests are dependent on the job though, so they can't always be broken into smaller pieces.

Part B: (20)

Consider the following reference string for a process: 1 2 3 4 3 4 3 4 3 4 5 6 5 6 5 6 4 5 6 1

What is the number of page faults with pure demand paging – with pre-allocation of frames.

6 - fault order: 1, 2, 3, 4, 5, 6 (if we consider the first load a fault as well)

What is the number of page faults when 1 page frame is allocated for the process?

20 – fault order: 1 2 3 4 3 4 3 4 3 4 5 6 5 6 5 6 4 5 6 1

What is the number of page faults when 3 page frames are allocated, and the page replacement if FIFO? Show the references that cause a fault.

1 is loaded	1
2 is loaded	1, 2
3 is loaded	1, 2, 3
1 is replaced by 4	2, 3, 4
2 is replaced by 5	3, 4, 5
3 is replaced by 6	4, 5, 6
4 is replaced by 1	5, 6, 1

7 faults (if we consider the first 3 loaded to be faults)

What is the number of page faults when 3 page frames are allocated, and the page replacement if LIFO? Show the references that cause a fault.

1 is loaded	1
2 is loaded	2, 1

3 is loaded	3, 2, 1
3 is replaced by 4	4, 2, 1
4 is replaced by 3	3, 2, 1
3 is replaced by 4	4, 2, 1
4 is replaced by 3	3, 2, 1
3 is replaced by 4	4, 2, 1
4 is replaced by 3	3, 2, 1
3 is replaced by 4	4, 2, 1
4 is replaced by 5	5, 2, 1
5 is replaced by 6	6, 2, 1
6 is replaced by 5	5, 2, 1
5 is replaced by 6	6, 2, 1
6 is replaced by 5	5, 2, 1
5 is replaced by 6	6, 2, 1
6 is replaced by 4	4, 2, 1
4 is replaced by 5	5, 2, 1
5 is replaced by 6	6, 2, 1

19 page faults

Part C: (20)

Memory access time is 100 nanoseconds. Page fault detection and replacement algorithm and page fetch requires 1 millisecond.

What is the effective access time (EAT) when the page fault rate is 0 %?

EAT = 1*.1 + 0*(1+.1) = .1 → 100 nanoseconds

What is the EAT when the page fault rate is 1 %?

EAT = .99*.1 + .01*(1+.1) = .11 → 110 nanoseconds