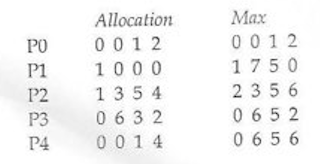
* **다음 문제에 대한 답을 서술해라.**

1. describe the detailed role of three Monitor queues (entry queue, condition queue, next queue)
2. x.wait를 semaphore로 구현하시오
3. bounded ring buffer를 producer / consumer 로 구현해라 (수도코드로)

* **뱅커스 알고리즘에 의해 지금 상태가 safe인지 확인**



available = (3,6,3,4)

* 1. **지금 상태가 safe한가? 맞다면 step by step으로 보여줘라**

P0-P2-P1-P3-P4

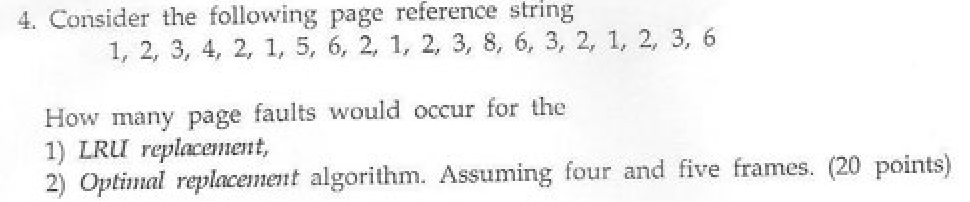
* 1. **p1에 (1,2,1,0)을 할당하려 한다. 가능한가?**

\*need보다 작거나 같은지 확인: P1’s need(0,7,5,0) 인덱스 0인 자원의 instance 수가 요구하는 것이 더 많아서 불가능하다.

\*available보다 작거나 같은지 확인: 현재의 available보다 모든 요구하는 instance가 작으므로 만족

* **1 2 1 3 1 4 5 6 1 8 이런 식으로 돼서 4프레임에서**

1. FCFS
2. LRU의 경우 page fault가 몇 번 발생하는가



* **hierarchical paging, hashed page table, inverted page table을 설명하시오**
* **다음 문제에 대한 답을 서술해라.**

1. File system의 access 방법 4가지를 설명하시오.
2. compile time, load time, execution time 때 address linking이 어떻게 되는가

컴파일러는 각 프로세스의 symbolic address를 relocatable address로 바인딩

링커나 로더가 relocatable addr을 absolute addr로 바인딩

instruction과 data가 메모리 addr로 바인딩 되는 것은 위 3가지 방법 중 하나로 실행

* **다음 문제에 대한 답을 서술해라**

1. segmentation with paging memory management scheme을 설명해라.
2. internal fragmentation 과 external fragmentation의 차이점에 대해 설명해라

2) x.wait를 semaphore로 구현하시오

// x.count = x.count + 1

IF next – count > 0 Then V(next) => next queue가 차있다면 그걸 body로

ELSE V(mutex) => 아니라면 entry queue에 있는 걸 body로

P(x-sem)

x.count = x.count – 1

x.wait뿐만 아니라 x.signal도 외우기

3) bounded ring buffer를 producer / consumer 로 구현해라 (수도코드로)

// 그냥 consumer하려는데 비어있으면 안되고, producer하려는데 차있으면 안되고 그런 것을 보여주면 될 듯

2. 뱅커스 알고리즘에 의해 지금 상태가 safe인지 확인 => 2009년 기말에 1번과 똑같음(available만 다르고)

alloc max available

P0 (0,0,1,2) (0,0,1,2) (3,6,3,4)

p1 (1,0,0,0) (1,7,5,0)

p2 (1,3,5,4) (2,3,5,6)

p3 (0,6,3,2) (0,6,5,2)

p4 (0,0,1,4) (0,6,5,6)

1) 지금 상태가 safe한가? 맞다면 step by step으로 보여줘라

// 해보면 safe함. work, need 그거 통해서 보여주기

2) p1에 (1,2,1,0)을 할당하려 한다. 가능한가?

// 이건 max – alloc이 need인데, 이 need에 1을 이미 넣을 수 없어서 불가능

3. 1 2 1 3 1 4 5 6 1 8 이런식으로 돼서 4프레임에서

1) FCFS

2) LRU의 경우 page fault가 몇 번 발생하는가

// 그림 그리면서 해보면 알 수 있음

4. hierarchical paging, hashed page table, inverted page table을 설명하시오

// 뭐 페이지 2개 쓰고 하는거 써주면 됨

5. 1) File system의 access 방법 4가지를 설명하시오.

// 이거 아마 파일 시스템 맨 첫장 오른쪽 아래 부분인가에 나옴

2) compile time, load time, execution time 때 address linking이 어떻게 되는가

// 이것도 파일 시스템 첫장 우측 위인가? 두 번째 장인가에 나옴

6. SCAN과 C-LOOK을 설명하시오.

// 이정도는 알아서