

# PA3: Virtual Memory Manager 201421023\_홍준업

---

## 인적사항

---

- 사이버보안학과
- 201421023
- 홍준업

## 과제 수행 정도

---

### 수행 완료

#### to\_phy\_addr 함수

- virtual address가 입력됐을때 physical memory의 address를 구해서 리턴
- bit masking을 통해 page number, offset을 구함
- lookup\_tlb 함수를 통해 tlb를 확인 (이후 수행은 tlb함수와 lookup\_page\_table함수에서 실행), 결과로 page frame을 얻음
- page frame과 offset의 shift 연산을 통해 physical address를 구함

#### lookup\_phy\_mem 함수

- physical address로 physical memory의 값을 얻어와서 리턴
- bit masking을 통해 frame number 와 offset을 구함
- 구한 frame number와 offset으로 physical memory에서 값을 구함

#### page\_fault\_handler

- tlb miss가 나고 page table에서 page가 존재하지 않을때 함수 실행
- backing store에서 요청된 page number의 페이지를 읽어 physical memory의 frame에 로딩
- page table에 frame number 기록

## 과제를 수행하면서 배운 것

---

### bit masking & shift 연산

- bit masking을 통해 원하는 위치의 비트만을 뽑아낼 수 있었다.
- 또한 shift 연산을 통해 자리 이동(곱셈, 나눗셈)을 할 수 있었다.

### 가상 메모리 동작

- virtual address에서 얻은 page number와 offset으로 physical address를 얻어오는 방식을 이해할 수 있었다.
- FIFO 방식의 replacement에서는 linear하게 접근하게 되면 매번 page fault가 날 수 밖에 없다는 것을 알 수 있었다.
- random 하게 메모리에 접근 했을때 tlb hit이 많이 나오는 것을 보아 더 빠르게 동작할 수 있을 것이라고 예상할 수 있었다.
- FIFO방식의 replacement에서도 tlb hit 비율이 높았는데 다른 알고리즘을 사용하면 더 빠르게 동작할 수 있을 것이라고 예상해 볼 수 있었다.

## 과제에 대한 피드백

---

- 메모리에 대한 이해가 되었으므로 다음에는 page replacement 알고리즘을 FIFO가 아닌 다른 것을 사용해서 시도 해보면 더 좋을 것 같다.