

운영체제 3 차과제

(1) 과제 설명

- **목표:** Priority Scheduler 구현 및 실험

- 0 단계: XV6 운영체제의 스케줄러인 void scheduler(void) 함수 분석하기
 - 분석내용 보고서에 추가

- 1 단계: 각 프로세스에 "priority" 개념 추가
 - proc.h 파일안에 있는 "proc" 구조체에 "int priority" 멤버 추가
 - 프로세스가 생성될 때, priority 값은 5 이어야 함 (최초로 실행되는 프로세스부터)
 - void userinit(void) 함수 와 static struct proc* allocproc(void) 함수 확인하여 구현
 - fork()를 통해서 프로세스가 생성될 때 자식 process 는 부모 process 와 같은 priority 값을 가져야함
 - "set_proc_priority" 시스템 콜 구현
 - process ID 와 priority 값을 arguments 로 사용하여 해당 ID 값을 가지는 process 의 priority 값 수정
 - priority 는 1 ~ 10 까지의 값을 가질 수 있음
 - 1 이 가장 높은 priority
 - 10 이 가장 낮은 priority 임
 - "get_proc_priority" 시스템 콜 구현
 - process ID 를 argument 로 사용하여 해당 ID 를 가지는 process 의 priority 값 출력

- 2 단계: void scheduler(void) 함수 수정하여 Priority Scheduler 구현
 - 선점형 방식의 Priority scheduling 시스템을 구현하고 실험

- 단, Starvation 이 절대로 발생하면 안되고, 이를 해결하기 위해서 모든 Process 가 scheduler 함수에서 선택되어 실행한 횟수를 기억하고, 이를 통해 해결할 것
 - starvation 현상이 나타나는 모습을 직접 관측해 볼 것
- 여러 개의 다양한 process 들을 생성하여 실험해 보고, 보고서 작성할 것

(2) 제출기한

2024 년 11 월 12 일 자정

(3) 제출방법

스마트캠퍼스를 통해서 제출

(4) 제출물

- 아래 파일들을 하나의 압축 파일로 제출 해야함.
- 압축파일의 이름은 반드시 "os3_학번.zip" 으로 해야함
 - 보고서 (반드시 자신의 글로 작성할 것)
 - 소속, 학번, 이름
 - 개발환경
 - 수정 및 작성한 소스코드에 대한 설명
 - 과제 수행 중 발생한 문제점과 해결 방법
 - 직접 수정하고 작성한 소스파일 전체 (XV6 소스코드 전체 올리지 말 것)
 - 직접 수정하고 작성한 파일들에 대한 간략한 설명을 "ReadMe.txt"의 파일이름의 텍스트 파일로 작성