



2016년 2학기 운영체제

Assignment 2

Dept. of Computer Engineering,
Kwangwoon Univ.

Requirements

▶ **Report**

- ▶ Introduction : 5줄 이하
- ▶ Reference : 각 과제 별
 - ▶ e.g. 친구 도움, 책, 인터넷 사이트 주소 등
 - ▶ 강의자료만 이용한 경우 생략 가능
- ▶ Conclusion

Requirements

▶ Source Code

▶ 2-1

- ▶ fork.c // fork를 이용하여 구현한 코드
- ▶ thread.c // pthread를 이용하여 구현한 코드
- ▶ Makefile // 두 application을 모두 컴파일 하도록 작성

▶ 2-2

- ▶ filegen.c // 테스트용 파일 생성 코드
- ▶ schedtest.c // 성능 측정 코드
- ▶ Makefile // 두 application을 모두 컴파일 하도록 작성

▶ Copy 발견 시 0점 처리

Requirements

▶ Softcopy

- ▶ 보고서 및 소스파일은 하나의 압축파일로 압축하여 제출 (tar.gz)
- ▶ 보고서는 PDF로 변환할 것
- ▶ **11월 4일 금요일 23:59:59** 까지, [U-캠퍼스] → [온라인참여학습] → [과제제출]
 - ▶ Due Delay 없음

▶ Hardcopy

- ▶ 제출하지 않음

Assignment 2-1

▶ 다음의 작업을 다중 프로세스/쓰레드로 수행하는 프로그램을 작성

- ▶ `fork()`/POSIX thread를 사용하여 여러 개의 프로세스/쓰레드가 아래의 코드를 동시 수행하도록 구현

```
for(i=0; i<1E9; i++)  
{  
    sum += i;  
}
```

▶ 프로세스/쓰레드 생성 시간, 전체 프로그램 수행 시간 측정

- ▶ 시간 측정 시 `clock_gettime()` 사용
- ▶ 각자 실험 환경에 맞춰 성능 차를 확실히 설명하기 위한 loop 횟수, 생성하는 프로세스/쓰레드 수 임의 조정 가능
- ▶ 결과에 대한 분석 내용에 따른 점수 차등화
- ▶ e.g. 다중 프로세스로 동작하는 프로그램 수행 결과

```
sslab@osdes:~$ ./fork  
Elapsed time after making processes: 194.18363583 sec  
Elapsed time to completion: 265.342905274 sec  
sslab@osdes:~$
```

Assignment 2-1

▶ 주의 사항

- ▶ 매 실험 전에 아래의 명령어를 수행할 것
 - ▶ 캐시 및 버퍼를 비워서 실험에 영향을 주는 요소를 제거

- ▶ `$ sync`
 - ▶ Linux command to flush file system buffer
- ▶ `$ echo 3 | sudo tee /proc/sys/vm/drop_caches`
 - ▶ Linux commands to free pagecache, dentries, and inodes

- ▶ 여러 번 실험하고, 그 값에 평균을 취할 것

Assignment 2-2

- ▶ CPU 스케줄링 정책이 다중 프로세스의 동작에 미치는 영향 분석
 - ▶ 프로그램 1. filegen (filegen.c)
 - ▶ Step 1. 특정 디렉토리("./temp")를 생성
 - ▶ Step 2. 이에 무작위의 integer 수($\geq 1,000,000,000$)가 기록되어 있는 파일 (./temp/0, ./temp/1, ./temp/2, ...)을 생성할 프로세스 수 만큼(MAX_PROCESSES) 생성

```
...
#define MAX_PROCESSES 10000
...
    for(i=0; i<MAX_PROCESSES; i++)
    {
        FILE *f_write = fopen(...
        fprintf(f_write, "%d", 1E9+rand()%1E9);
        fclose(f_write);
        ...
    }
...
```

Assignment 2-2

- ▶ **CPU 스케줄링 정책이 다중 프로세스의 동작에 미치는 영향 분석**
 - ▶ 프로그램 2. schedtest (schedtest.c)
 - ▶ Step 1. **MAX_PROCESSES** 만큼 프로세스를 생성
 - ▶ Step 2. 각 프로세스에서 CPU 스케줄링 정책을 변경 (Hint. sched_setscheduler())
 - ▶ Step 3. 각 i (단, $0 \leq I < \text{MAX_PROCESSES}$) 번째 프로세스에서 미리 만들어져 있는 i 번째 파일(./temp/i)에서 integer 데이터 읽음
 - ▶ 프로그램 1을 수행한 후, 프로그램 2를 CPU 스케줄링 정책을 바꿔가면서 수행
 - ▶ CPU 스케줄링 정책이 프로세스 동작 시간에 미치는 영향을 분석

Assignment 2-2

▶ 주의 사항

- ▶ 매 실험 전에 아래의 명령어를 수행할 것
 - ▶ 캐시 및 버퍼를 비워서 실험에 영향을 주는 요소를 제거

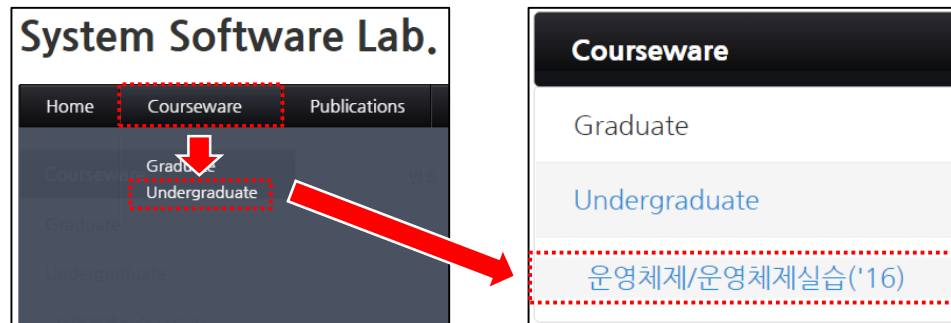
- ▶ `$ sync`
 - ▶ Linux command to flush file system buffer
- ▶ `$ echo 3 | sudo tee /proc/sys/vm/drop_caches`
 - ▶ Linux commands to free pagecache, dentries, and inodes

- ▶ 여러 번 실험하고, 그 값에 평균을 취할 것

과제 질문 시 Q&A 게시판 사용

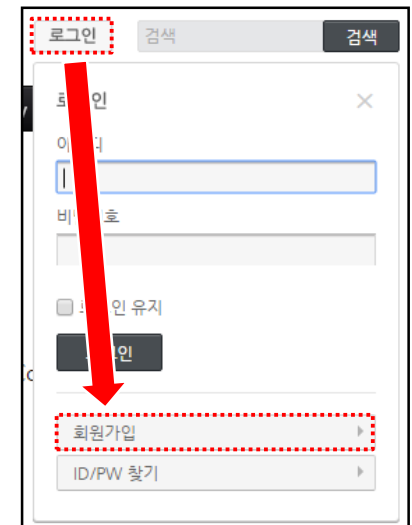
▶ 과제 질문은 **모두 아래의 홈페이지에서만** 받음

▶ sswlab.kw.ac.kr → Courseware → Undergraduate → "운영체제/운영체제실습('16)"



▶ 게시판을 사용하기 위해서 홈페이지 가입이 필요

- ▶ 화면 오른쪽 상단의 "로그인" → "회원가입"
- ▶ 아래의 조건이 충족 되지 않을 경우,
질문에 답변이 거부되거나 질문 글이 삭제될 수 있음
 - ▶ "닉네임"이 본인 실명이 아닌 경우
 - ▶ 기타 회원 가입 시 입력한 내용이 올바르지 않은 경우



과제 질문 시 Q&A 게시판 사용

• 과제 질문은 **모든 아래의 홈페이지에서만** 가능

과제 마감 당일(금요일)은
오후 5시 이전에 올라온 질문에 한하여 답변
(‘2’에도 동일하게 적용)

홈페이지에 **문제가 있어 질문이 불가능한 경우,**
담당 조교에게 직접 연락
→ 김규식 조교, kks@kw.ac.kr

• 아래의 조건이 충족되지 않을 경우

질문 시, 본인이 질문하는 과제를 명시할 것
(ex. 이론 수업의 2차 과제, 1번 ...)