

Lab. 9

# Memory Mapping

Duksu Kim



# 공통사항

- 모든 System call 및 Standard C function 사용 가능
  - 단, 외부 라이브러리 사용 불가
- Source code 및 실행파일 이름은 문제 번호 사용
  - 예) 9\_1.c 및 9\_1.out 등
- 시간 측정 필요 시, 자신이 만든 측정 함수 사용
- EL 제출 시,
  - 모든 source code를 모은 한글/워드 파일 별도 제출



Class 01

# Type A

1 조

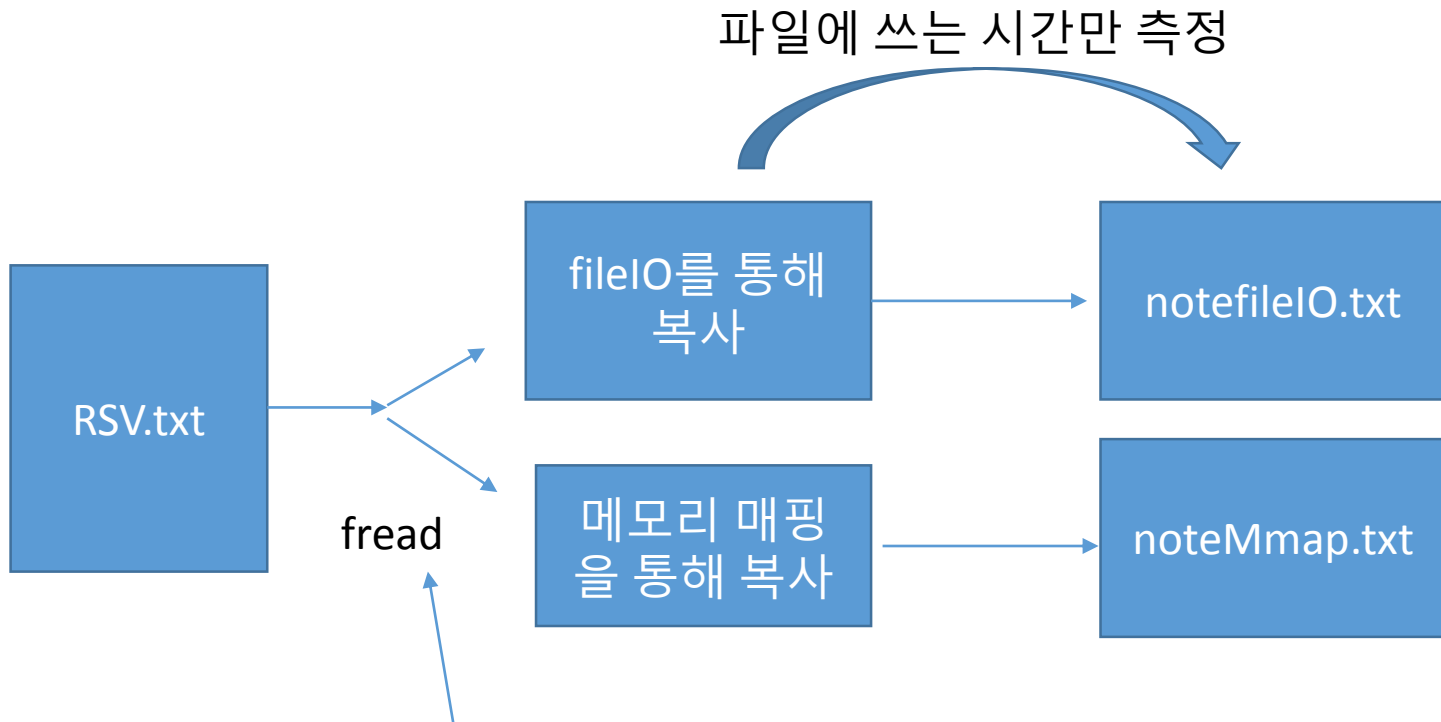
이선훈 정준욱 한동곤



# Lab 9-1. 텍스트 문서 복사하기

## • 문제 설명

- 텍스트문서를 복사하는 프로그램을 만들고 이때 걸리는 시간을 측정한다.



버퍼크기는 자신의 **페이지크기**만큼



# Lab 9-1. 텍스트 문서 복사하기

## • 문제 설명

- 읽을 RSV.txt파일을 제공해드립니다.
- 복사한 파일의 이름은 임의로 설정하시면 됩니다.
- 읽는 건 파일 입출력으로 합니다. 파일 입출력은 버퍼를 사용하는데 이 버퍼의 사이즈를 페이지 사이즈로 설정합니다.
  - 페이지 사이즈로 설정하지 않으면 시간차가 나지 않을 수 있습니다.
- 결과는 반드시 메모리 매핑이 파일 입출력보다 빨라야 합니다.
- 시간 측정은 반드시 파일을 쓰는 부분에서만 시행합니다.

```
SHTimer_on(0);  
fprintf(fpW, "%s", buffer);  
SHTimer_off(0);
```

이전 과제의 타이머를 쓰셔도 됩니다.



# Lab 9-1. 텍스트 문서 복사하기

- **Hints**

- `ftruncate` 쓰셔야 파일에 제대로 동기화 됩니다.
- 메모리 매핑으로 복사할 땐, 메모리의 내용을 파일에 쓰는 거라서 `reference copy`하시면 안됩니다.



# Lab 9-1. 텍스트 문서 복사하기

- 실행의 예

```
lsh@DESKTOP-AJ8HLSI:~/Lab8$ gcc -o Lab9_1.out SHTimer.c SHTimer.h Lab9_1.c
lsh@DESKTOP-AJ8HLSI:~/Lab8$
lsh@DESKTOP-AJ8HLSI:~/Lab8$ ./Lab9_1.out
0th timer is fileI0
1th timer is mmap
0th timer sec :0.225(ms)
1th timer sec :0.060(ms)
```

```
lsh@DESKTOP-AJ8HLSI:~/Lab8$ ls -l
total 224
-rw-rw-rw- 1 lsh lsh 1203 Nov 16 20:09 Lab9_1.c
-rwxrwxrwx 1 lsh lsh 13824 Nov 16 20:19 Lab9_1.out
-rw-rw-rw- 1 lsh lsh 36864 Nov 16 20:27 noteFI0.txt
-rw-r--r-- 1 lsh lsh 102400 Nov 16 20:27 noteMmap.txt
-rw-rw-rw- 1 lsh lsh 36521 Nov 16 20:06 RSV.txt
-rw-rw-rw- 1 lsh lsh 2675 Nov 15 23:27 SHTimer.c
-rw-rw-rw- 1 lsh lsh 137 Nov 15 23:27 SHTimer.h
```



# Lab 9-2. 한컴타자연습

## • 문제 설명

- Producer와 consumer 가 있다.
- Consumer가 시작신호를 보내면 Producer에서는 주어진 3000개의 단어장에서 단어 100개를 랜덤으로 뽑아 출력한다.
- Consumer는 scanf로 올바른 단어를 typing한다
- 출력된 단어장에서 그 위치의 단어가 사라진다.
- 스코어가 증가한다.





# Lab 9-2.

- Hints
- 문자열 동적배열 하는법 strcpy 사용
- 단어가 사라진다 = 그 위치 단어 = “ ”;
- Mmap으로 서로 통신
- 두 문자열이 같음 strcmp



# Lab 9-2.

## • 실행의 예

```
diffrent.txt      Lab2_3_message.txt      Lab6_1.out      Make_h1.out
system@DESKTOP-5A7TQFA: ~/2013136116$ ./temp.out
```

system@DESKTOP-5A7TQFA: ~/2013136116

```
system@DESKTOP-5A7TQFA: ~$ cd 2013136116
system@DESKTOP-5A7TQFA: ~/2013136116$ ./tempc.out
are you want start? [y / n]
```



# Lab 9-2.

## • 실행의 예

```
=====
limited      city      week      field
white       competitor  lose      liberal
chicken     degree      dirt      disappear
suicide     external    environmental exceed
result      limit       priority  normally
subsequent  invite      glass     why
mouth       noise       inner     desire
establishment minute     consumer some
somebody    coast       commit    toy
industrial  proper     meet      official
apartment   shoulder   while     tobacco
carry       grant      female    you
too         chest      purpose   tendency
replace     billion    wire      hi
primarily   moral      self      boyfriend
broad       whenever  appointment arrangement
direct      cup        wet        soon
aircraft    PC         honor     consumption
hunting     lots       cultural  privacy
transfer    print      prime     very
valley      gaze       stage     privacy
jail        unlike    borrow    eager
leading     temporary learning  relevant
wide        minister  setting   film
terms       tank      component among
score = 0
```

```
system@DESKTOP-5A7TQFA: ~/2013136116
system@DESKTOP-5A7TQFA:~$ cd 2013136116
system@DESKTOP-5A7TQFA:~/2013136116$ ./tempc.out
are you want start? [y / n]
y
print answer!
```

Consumer 가 올바른 답을 보내면  
producer에서 그 위치에 정답이 사라지게  
하면 된다.



# Lab 9-2.

## • 실행의 예

```
=====
limited      city      week      field
white       competitor lose      liberal
chicken     degree    dirt      disappear
suicide     external  environmental exceed
                                     normally
subsequent  invite    glass
mouth       noise
establishment consumer
somebody
industrial  proper    meet      official
shoulder   shoulder while    tobacco
grant      chest     purpose   you
too        billion  wire      tendency
replace    moral    self      hi
primarily  whenever appointment boyfriend
broad      cup      arrangement
direct
aircraft   honor    consumption
hunting    cultural
transfer   print
valley     gaze
jail       unlike
           temporary
           minister
           learning
           setting
           component
score = 15
```

system@DESKTOP-5A

```
print answer!
limit
print answer!
inner
print answer!
why
print answer!
wet
print answer!
prime
print answer!
PC
print answer!
lots
print answer!
female
print answer!
minute
print answer!
coast
print answer!
apartment
print answer!
leading
print answer!
tank
print answer!
commit
print answer!
```



Class 02

# Type B

1조

최문혁 조용태 정우진 임상훈



# Lab 9-1. Up & Down Game

## • 문제 설명

- 목적 : 프로세스간 메모리 공유와 rand() 함수 활용 해보기
- ✓ Answer.o : com1, com2가 서로 맞추어야 할 숫자를 생성 & 출력 후 종료.
- ✓ com1.o : com2의 숫자를 일정 간격으로 제시한다.
- ✓ com2.o : com1의 숫자를 일정 간격으로 제시한다.
- ✓ 두개의 com중 먼저 맞춘 com이 있다면 두개의 com은 end.
- ✓ 모든 동작은 프로세스들이 자동으로 동작하도록 하십시오.



# Lab 9-1. Up & Down Game

## • 실행의 예

```
선택 system@DESKTOP-VLU0DET: ~/temp
system@DESKTOP-VLU0DET:~/temp$ ./producer.out
Producer : 59 Up!
Producer : 71 Down!
Producer : 69 Down!
Producer : 64 Up!
Producer : 65 Up!
Producer : 67 Down!
66 Great!
system@DESKTOP-VLU0DET:~/temp$

system@DESKTOP-VLU0DET: ~/temp
system@DESKTOP-VLU0DET:~/temp$ ./answer.out
producer answer : 99
consumer answer : 66
system@DESKTOP-VLU0DET:~/temp$ ./consumer.out
Consumer : 5 Up!
Consumer : 75 Up!
Consumer : 87 Up!
Consumer : 90 Up!
Consumer : 93 Up!
Consumer : 100 Down!
Consumer : 98 Up!
system@DESKTOP-VLU0DET:~/temp$
```

- ✓ 정답이 66이고,
- ✓ 만약 com0이 59(Up)와 71(Down)을 순서대로 제시했을 때, com은 다음 제시 값을 60과 70사이에서 제시해야 합니다.



# Lab 9-2. 숫자 야구 Game

## • 문제 설명

➤ 목적 : 프로세스간 메모리 공유 익히기!

✓ 3자리 숫자야구게임 입니다.

✓ com process는 3자리 숫자를 랜덤 생성 & player process가 제시한 답을 확인하여 strike, ball을 알려줍니다. ( =>자신의 웰에 표시 )

✓ Player process는 3자리 숫자를 입력 받습니다.

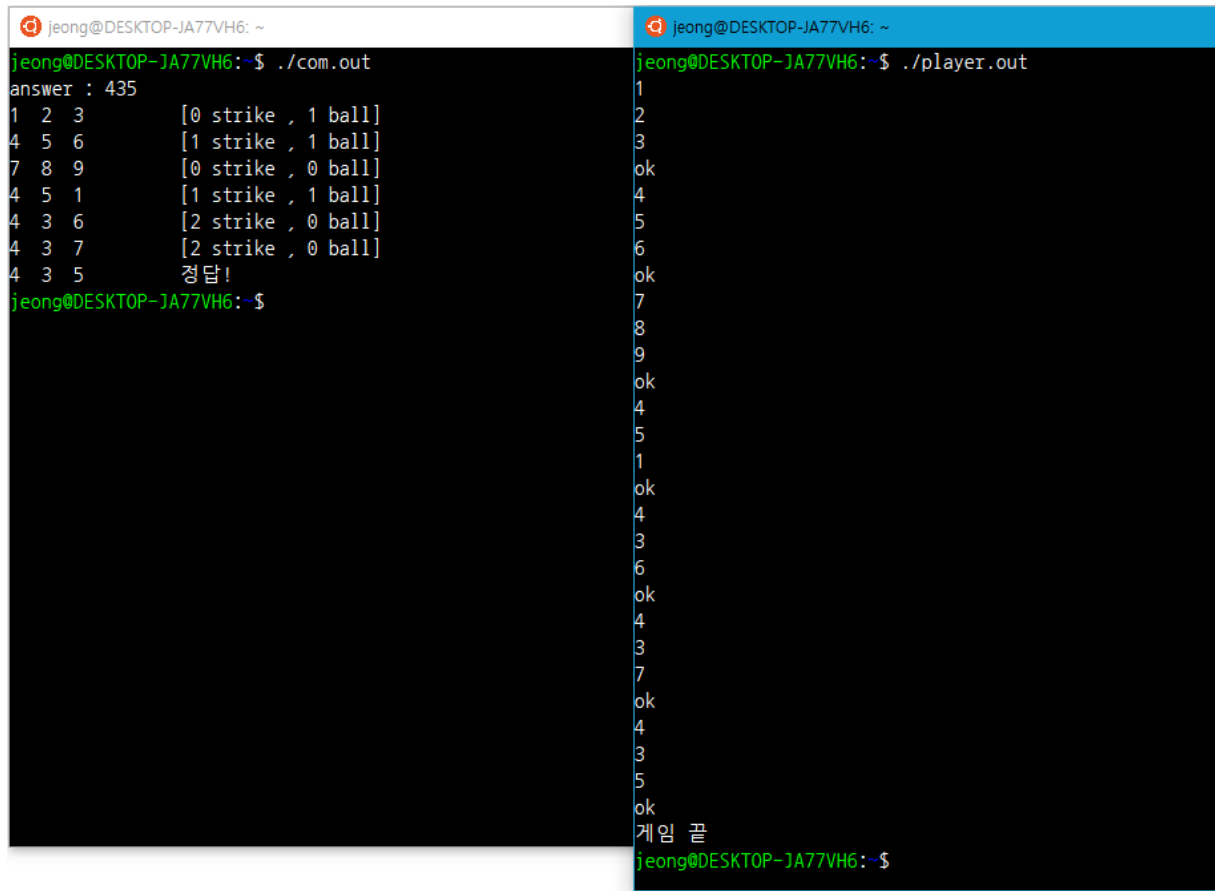
✓ 3Strike일 경우 com, player process는 동시 종료합니다.





# Lab 9-2. 숫자 야구 Game

- 실행의 예



```
jeong@DESKTOP-JA77VH6: ~  
jeong@DESKTOP-JA77VH6:~$ ./com.out  
answer : 435  
1 2 3      [0 strike , 1 ball]  
4 5 6      [1 strike , 1 ball]  
7 8 9      [0 strike , 0 ball]  
4 5 1      [1 strike , 1 ball]  
4 3 6      [2 strike , 0 ball]  
4 3 7      [2 strike , 0 ball]  
4 3 5      정답!  
jeong@DESKTOP-JA77VH6:~$  
  
jeong@DESKTOP-JA77VH6: ~  
jeong@DESKTOP-JA77VH6:~$ ./player.out  
1  
2  
3  
ok  
4  
5  
6  
ok  
7  
8  
9  
ok  
4  
5  
1  
ok  
4  
3  
6  
ok  
4  
3  
7  
ok  
4  
3  
5  
ok  
게임 끝  
jeong@DESKTOP-JA77VH6:~$
```



Common Problem

# Lab 9-3

By DS Kim



# Lab 9-3. Do Not Disturb Mode

- SIGUSR1을 이용한,  
프로세스의 방해 금지 모드 설정/해제
- 방해 금지 모드
  - 모든 SIGNAL을 blocking (SIGKILL, SIGSTOP은 제외)
  - 방해 금지 모드 해지 시,
    - 방해 금지 모드 중 받은 signal을 화면에 출력
- Hints : `signal(2)`, `sigprocmask(2)`, `signal set`



# Lab 9-3. Do Not Disturb Mode

## • 실행의 예

```
$ ./9_3.out
```

```
Do Not Disturb mode on!
```

```
Do Not Disturb mode off!
```

```
[During the DND mode]
```

- Received signal: Hangup
- Received signal: Interrupt
- Received signal: Quit



```
$ kill -10 2542
```

```
$ kill -1 2542
```

```
$ kill -2 2542
```

```
$ kill -3 2542
```

```
$ kill -10 2542
```



Extra Problem

# Lab 9-X

By DS Kim



# Parallel Matrix Multiplication

- **행렬 곱을 구하는 병렬처리 프로그램 작성**
- **입력 : Matrix A, Matrix B, Output file, # of threads**
  - Matrix A, B는 파일 형태로 입력
    - Matrix size = 1024 by 1024
    - Rand 함수를 통해 임의 생성 후, 파일에 저장 후 사용
- **출력 : 행렬 곱 결과**
  - 결과는 하나의 파일(Output file)에 저장 될 것
- **N개의 thread에 일을 균등하게 배분 할 것**
- **Hints : Memory mapping**
- **Extra point : +1 (질문 포인트)**

