



김현수 (hscornelia) 수정

팔로우 중인 프로필 전체 프로필 보기

[돌소리](#)
[알립니다](#)
[구인구직란](#)
[18학번 모임](#)
[버그 제보 및 기능 제안](#)
[김현수](#)
[족보](#)

개인정보 처리방침

[2015-2] 시스템프로그래밍 (에거 교수님) 중간고사 족보

6년 전



신동진

주관적인(?) 복기

필기(80min)

1. 객관식 16문제 (0.5pt each) (주의사항 : Mark all ~이 답이 한개인 경우가 많음)

1. When returning from a syscall `l_return` is set to

- a. the previous instruction
- b. current instruction
- c. next instruction
- d. kernel doesn't return

2. When returning from recoverable page fault

- a. the previous instruction
- b. current instruction
- c. next instruction
- d. kernel doesn't return

3. When returning from illegal memory access

- a. the previous instruction
- b. current instruction
- c. next instruction
- d. kernel doesn't return

4. While exec an exception handler, processor runs in
 - a. stack mode
 - b. kernel mode
 - c. user mode
 - d. mmu mode
5. exception cause by the timer interrupt is
 - a. synch
 - b. asynch
6. Exception cause by syscall
 - a. synch
 - b. asynch
7. exception caused by an illegal memory access
 - a. synch
 - b. asynch
8. when invoking a syscall on linux args of calls via
 - a. stack
 - b. reg
 - c. ~
9. Mark all signals can neither be caught nor ignored
(객관식입니다 보기가 생각안나서 그만)
10. What is the child process that has not been reaped
11. Mark all syscalls that are called once but returns twice
12. Mark all syscalls that never returns
13. signal sent from parent process to child process
 - a. goes through kernel
 - b. directly ~
14. signals received by a process
(handling issue)
15. Where are the page table located
16. Mark all options that cannot use anonymous shared memory for interprocess communication
 - a. parent-child
 - b. sibling-sibling
 - c. unrelated processes
 - d. parent-grandchildren

2. PIC. 서술형 (8pt)

- A. Explain the difference between absolute and PC-relative relocation
- B. PIC uses which form of addressing?
- C. In what common situation in the linux OS is PIC useful?
 - shared libraries
 - address randomization
 - running several programs on system that do not support virtual memory
- D. When generating PIC what is the main difficulty on IA32?
 - IA32 does not allow access to the EIP directly which complicates position independent accesses to global data

3. Symbol Resolution (22pt)

- A. symbol 표 채우기
- weak/strong symbol
- type(external, global, local, -)
- section(.data, .text, .bss, -)
- Remarks(for - symbols)

B. 표 채우기 : Relocation / Byte Position / Type / Reference

C. Compute the value of each relocation in the linked executable.

4. VMM 서술형 (12pt)

- A. Explain why nested PT are necessary for 64-bit address space
 - without each PT would require $2^{64}/4KB * 8byte = 32PB$
- B.
 - (a) what is the minimal amount of memory required by the pPT to map the virtual address space 1000-2fff, 8000-8fff
 - (b) How many Physical pages are required at most to map the VA space?
 - 7 pages tables with each 28KB in total
- C. What is the maximum number of memory accesses required to translate a VA into a PA? explain
 - 4 Access (miss in TLB, for entries in each of 4 level)
- D. What is the name of the HW block ~ (write fullname)
 - TLB(Translation Lookaside Buffer)

5. (12pt)

```
one.c : int A[N] = {1}; (global)
two.c : int A[N]; (global)
three.c : int A[N] = {1}; (local)
four.c : int *A = calloc(N, 4);
```

A. section of A[N]?

```
one : .data
two : .bss
three : stack
four : heap
```

B. program size?

- one >> three = four > two (one >> three = four = two)

C. program runtime(slowest)?

- one > three = four > two (one > three = four = two)

6. VA translation (TLB, PTE 등등 주고 VA->PA 하는 굵직한 문제) (12pt)

7. 실행결과를 쓰시오 (12pt)

pid_t pid;

void f(void)

```
{  
    printf("{1}\n");  
}
```

void g(void)

```
{  
    printf("{2}\n");  
}
```

void h(int sig)

```
{  
    printf("{3}\n");  
    kill(pid, SIGUSR1);  
}
```

void i(int sig)

```
{  
    printf("{4}\n");  
    exit (0);  
}
```

void main() {

```
    signal(SIGUSR1, h);  
    atexit(f);
```

```
    printf("{5}\n");
```

```
    if ((pid = fork()) == 0) {  
        signal(SIGUSR1, i);  
        kill(getppid(), SIGUSR1);  
        while (1) {};  
    } else {  
        atexit(g);  
        waitpid(pid, NULL, 0);  
    }
```

```

    }
    printf("{6}\n");
}

```

실기 (90min)

1. 유사 버퍼랩 문제

계산기 프로그램이 있음

명령행 입력에 따라 add와 mul이 실행가능한데 div는 정상적으로 진입불가

A. div에 진입하게 입력을 넣으시오

B. A를 간단하게 하면 세그폴로 종료되는데, 이를 깔끔하게 종료되게 입력을 넣으시오.

2. Library Interposition

A. malloc, free, realloc, calloc을 실행마다 크기/횟수를 카운팅해서 프로그램 종료시 로그를 출력하도록 (빠대
가 웬만큼 있고 주어진 함수+매크로를 조합해서 짜면 됨. 핵심은 dlsym 사용) 싹참

B. exit without free 감지

C. double free 감지, 인터셉트

Bonus. 워더라

Etc.

필기 평균 : 38.66/80 | 실기 평균 : 13.59 / 100 | 총평균 : 41.38 / 100

총점 = 필기 80% + 실기 20%

필기 채점기준이 자비로운 편 + 부분점수 후함

실기 시간이 매우 모자람 (1.A B 2.A B C Bonus 순으로 점수평균 : 10/20 6.25/10 4.95/20 10/30 1.67/10 0/1
0)

실기에 해결여부와 별도로 코드점수가 있는데 예외처리(null 체크)등을 봄. 은근 비중이 있음.

실기는 올해부터 처음 하는거라 많이 달라질 가능성이 있음

필기와 실기 중간에 교수님이 저녁으로 파파존스 시켜주심

총점 30점 이하는 성적향상을 위한 면담의 기회가 주어짐

크아아앙

--

RevreServeR Owner

추천(0)

▶ 시스템프로그래밍 × 태그추가

L 4개 더 보기



황재동

6년 전

필기 자비롭고 후한데도 평균이 ㄸ

추천(0)

답글

확인