

시스템 소프트웨어

HW2

Moon Gi Seok

숙제 내용

- 다음 슬라이드에 나오는 5문제를 푸시오.

- ▶ 문제 1,2,3,5 는 실습 문제

- C언어를 실습 서버에 원격 접속해서 개발하고, gcc 명령어를 이용해 컴파일 및 실행해보자.

- ▶ 문제 4는 텍스트로 답변 가능

- 제출 파일:

- ▶ 모든 내용을 통합해서 하나의 pdf 파일로 제출

- 압축파일 및 기타 파일 업로드 금지

문제 1.

- C 코드를 작성하고 실행하여 현재 실습 머신이 리틀 엔디안인지 확인하고, 코드와 실행 결과 캡처를 첨부하세요.
 - ▶ 짧게 2줄로 설명 추가

문제 2

- 다음을 지원하는 c 코드를 작성하고, 실행 결과를 출력하시오
 - ▶ `int a = -9;` 일 때, 변수 `a`의 MSB(Most Significant Bit)가 1인지 확인
 - ▶ `a/8` 와 `a >> 3`가 같은지를 확인
 - 결과에 대한 설명 짧게 3줄 이내로 설명하시오.

문제 3

- 아래 속성 중에, **overflow가 발생**하면 만족하지 않는 상황이 발생할 수 있다.

```
int x = foo();  
int y = bar();  
unsigned ux = x;  
unsigned uy = y;
```

- $x < 0 \Rightarrow ((x*2) < 0)$
- $ux \geq 0$
- $x \& 7 == 7 \Rightarrow (x \ll 30) < 0$
- $ux > -1$
- $x > y \Rightarrow -x < -y$
- $x * x \geq 0$
- $x > 0 \&\& y > 0 \Rightarrow x + y > 0$
- $x \geq 0 \Rightarrow -x \leq 0$
- $x \leq 0 \Rightarrow -x \geq 0$

- 대표적으로 1) $x < 0 \Rightarrow x^2 < 0$, 2) $x*x \geq 0$, 3) $x > 0 \&\& y > 0 \Rightarrow x+y > 0$
- 여기서, 첫번째와 두번째 property의 반례와 실행결과를 살펴보자 (다음 슬라이드에 포함됨)
- 마지막 세번째 property ($x > 0 \&\& y > 0 \rightarrow x + y > 0$) 이 성립하지 않는 반례를 찾아, 코드로 작성하고 실행 결과를 보여주세요.

문제 3 (Cont.)

- $x < 0 \Rightarrow x^2 < 0$

```
#include <stdio.h>
#include <limits.h> // INT_MIN과 같은 상수를 사용하기 위해 필요

int main() {
    int x = INT_MIN; // 가장 작은 음수 값
    int b = x * 2;

    printf("x = %d\n", x);
    printf("x * 2 = %d\n", b);

    if (x < 0) {
        printf("x는 음수입니다.\n");
    } else {
        printf("x는 음수가 아닙니다.\n");
    }

    if (b < 0) {
        printf("x * 2는 음수입니다.\n");
    } else {
        printf("x * 2는 음수가 아닙니다.\n");
    }

    return 0;
}
```

```
ngseok@linuxserver1: /sys
x = -2147483648
x * 2 = 0
x는 음수입니다.
x * 2는 음수가 아닙니다.
```

- $x < 0 \Rightarrow x^2 < 0$

```
#include <stdio.h>
#include <limits.h> // INT_MAX 등을 사용하기 위해 필요

int main() {
    int x = 50000; // 큰 값으로 설정하여 오버플로우 유도
    int b = x * x;

    // x * x를 출력
    printf("x = %d\n", x);
    printf("x * x = %d\n", b);

    if (b >= 0) {
        printf("x * x는 양수입니다.\n");
    } else {
        printf("x * x는 음수입니다. (오버플로우 발생)\n");
    }

    return 0;
}
```

```
x = 50000
x * x = -1794967296
x * x는 음수입니다. (오버플로우 발생)
```

문제 4.

- 아래의 항목이 항상 성립하는지 설명하시오.

- ▶ 만족하지 않으면 반례만 제시

- $(x|-x)>>31 == -1$

- $ux >> 3 == ux/8$

$$x \& (x-1) != 0$$

문제 5

- 다음 슬라이드에 나오는 리눅스 커맨드를 실습 서버에 접속해서, 활용해보고 결과를 캡처해보자

문제 5 – 리눅스 커맨드

- **man (Manual Pages):**

- ▶ The ultimate reference guide built into your system.
- ▶ Type `man <command_name>` (e.g., `man ls`) to get detailed instructions and explanations on any command.

- **echo:**

- ▶ A versatile tool that prints text to the terminal. Use it for testing, displaying variables, or as part of more complex scripts. For example, `echo "Hello, Linux world!"`

- **clear:**

- ▶ Need a clean slate? This command wipes your terminal window clean.

- **history:**

- ▶ Displays a list of your previously used commands. This is invaluable for recalling commands and finding patterns in your usage.

문제 5 – 리눅스 커맨드

- **mkdir (Make Directory):**
 - ▶ Your tool for creating new directories (folders). Example: mkdir Documents
- **rmdir (Remove Directory):**
 - ▶ The counterpart to mkdir, this command deletes empty directories. (For directories with files inside, you'll need the more powerful rm)
- **touch:**
 - ▶ Creates an empty file or updates the timestamp of an existing file. Try touch newfile.txt
- **cat (Concatenate):**
 - ▶ A multi-purpose command frequently used to display the contents of files. For instance, cat shoppinglist.txt
- **cp (Copy): Creates copies of files.**
 - ▶ Example: cp report.txt report_backup.txt
- **mv (Move): Moves or renames files and directories.**
 - ▶ Use it like mv oldfile.txt newfile.txt or mv Documents/report.txt Documents/backup/