**REPORT**

**[Hw 2]**

원, 폰트, 동전이(가) 표시된 사진

자동 생성된 설명

**과 목 : 시스템소프트웨어03**

**담당교수 : 석문기 교수님**

**학 과 : 컴퓨터공학과**

**학 번 : 2021111971**

**이 름 : 이재혁**

텍스트, 클립아트이(가) 표시된 사진

자동 생성된 설명

**문제1**

**소스코드**

#include <stdio.h>

typedef unsigned char \*pointer;

// 데이터의 주소를 가리키는 데이터 타입

void show\_bytes(pointer start, size\_t len) {

size\_t i;

for(i = 0; i < len; i++) {

printf("%p\t0x%.2x\n",start+i,start[i]);

printf("\n");

}

// 데이터 타입의 길이 만큼 낮은 주소부터 바이트에 저장된 값 출력

}

int main() {

int a = 15213;

printf("a = %x;\n", a);

show\_bytes((pointer)&a, sizeof(int));

}

**텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

**결과분석**

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

a의 낮은 주소와 값을 출력할 때 하위 8비트가 먼저 저장되어 있는 것을 확인할 수 있습니다. 즉, Little Endian 방식으로 값을 저장합니다.

**문제2**

텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

**소스코드**

#include <stdio.h>

int main() {

int a = -9;

// 1을 4byte의 최상위 위치인 32번째 칸으로 이동

// & 연산시 a의 최상위 비트의 결과가 출력된다.

if (a & (1 << 31)) {

// MSB == 1 : 음수

printf("MSB : 1 (음수)\n");

} else {

// MSB == 0 : 양수

printf("MSB : 0 (양수)\n");

}

printf("a / 8: %d\n", a / 8);

printf("a >> 3: %d\n", a >> 3);

// division과 shift의 연산결과를 출력

if (a / 8 == a >> 3) { // 두 연산을 비교

printf("a / 8 and a >> 3 are the same.\n");

} else {

printf("a / 8 and a >> 3 are different.\n");

}

}

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

**결과 분석**

**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명**

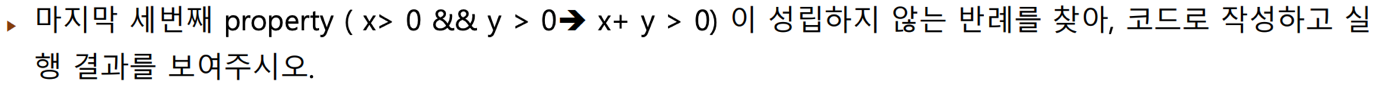
1을 31번 shift연산 후 a와 & 연산 한 결과 -> 최상위 비트 1 확인

a / 8 = -9/8 = (int)-1.125 -> -1 0에 가깝게 rounding 발생

a >> 3

* 1111 1111 1111 1111 1111 1111 1111 0111
* 1111 1111 1111 1111 1111 1111 1111 1110 = -2

**문제3**



**소스코드**

#include <stdio.h>

#include <limits.h> // INT\_MAX 사용

int main() {

int x = INT\_MAX; // 가장 큰 int (양수)

int y = 1; // 1 양수

// 변수가 양수인지 확인

printf("x : %d\n", x);

printf("y : %d\n", y);

if(x + y > 0) {

printf("x + y > 0 입니다.\n%d\n", x+y);

}

else {

printf("x + y < 0 입니다.\n%d\n", x+y);

}

}

**텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

**결과분석**

****

x : 0111 1111 1111 1111 1111 1111 1111 1111 > 0

y : 0000 0000 0000 0000 0000 0000 0000 0001 > 0

x + y : 1000 0000 0000 0000 0000 0000 0000 0000 < 0 (overflow 발생)

**문제4**

텍스트, 폰트, 스크린샷, 화이트이(가) 표시된 사진

자동 생성된 설명

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

* x가 0 일 때 x | -x 는 32개의 비트가 모두 0이다. 따라서 31번 shift연산을 시행해도 0 이다.

x & (x-1) != 0

* x가 8일 때
* x : 0000 0000 0000 0000 0000 0000 0000 1000
* x - 1 : 0000 0000 0000 0000 0000 0000 0000 0111
* x & (x-1) = 0

**문제5**

리눅스 서버 command 실습

|  |  |  |
| --- | --- | --- |
| man | 명령어에 대한 상세한 설명 | 텍스트, 스크린샷, 소프트웨어, 컴퓨터이(가) 표시된 사진  자동 생성된 설명 |
| echo | terminal에 text 출력 |  |
| clear | terminal 내용 지우기 | 텍스트, 스크린샷, 소프트웨어, 컴퓨터이(가) 표시된 사진  자동 생성된 설명 |
| history | 이전에 작성한 command 출력 |  |

|  |  |  |
| --- | --- | --- |
| mkdir | Directory 생성 |  |
| rmdir | Directory 삭제 |
| touch | 새로운 파일 생성 or 파일의 time stamp 갱신 |  |
| cat | 주로 파일 내용 확인에 사용 |  |
| cp | 파일 내용을 새로운 파일에 복사 |  |
| mv | 파일의 디렉토리 혹은 이름 변경 |  |