

Unix 프로그래밍 및 실습

2장. 파일 입출력

강의 내용

- ▶ 시스템콜을 이용한 파일 입출력
 - ▶ 강의 내용 (1 ~ 25/51)
 - ▶ <http://lily.mmu.ac.kr/lecture/13u2/ch02.pdf>

과제 (1)

- ▶ [예제 2-1], [예제 2-2], [예제 2-3] (각 10점씩)
 - ▶ 과제 개요 (2줄 이상)
 - ▶ 프로그램
 - ▶ 실행화면 캡처
- ▶ 응용 #1 (100점)
 - ▶ 파일을 5개 이상을 임의로 open하고, close 하면서 파일 이름과 fd 출력하기
- ▶ [예제 2-4], [예제 2-5] (각 10점씩)
- ▶ 응용 #2 (200점)
 - ▶ 명령행 (`main(int argc, char* argv[])`)을 이용하여 원본 파일 이름과 복사본 파일 이름을 전달받아 파일을 복사하는 프로그램을 작성하시오.
 - ▶ 원본 파일이 없거나 권한이 없는 경우 오류 메시지 출력
 - ▶ 기존에 복사본 파일이 있는 경우 덮어쓸 것인지 여부 확인하여 처리
- ▶ [예제 2-6], [예제 2-7], [예제 2-8] (각 10점씩)
- ▶ [예제 2-9], [예제 2-10] (각 10점씩)

과제 (2)

▶ 응용 #3 (700점)

▶ 숙박카드 구조체

```
typedef struct {  
    char        name[NAME_SIZE];  
    int         no_of_guests;  
} guest_card;
```

▶ 호텔 객실 번호는 1호실부터 순서대로 부여 (호실 제한 없음)

▶ 입실 기능 (100점)

- ▶ 사용자로부터 원하는 객실 번호를 입력 받아 현재 투숙 중인지 확인하여 빈 방이면 손님 이름과 투숙 인원 저장
- ▶ 이미 손님이 있는 방이면 객실 번호를 다시 입력 받아 진행

▶ 퇴실 기능 (100점)

- ▶ 사용자로부터 객실 번호와 이름을 입력 받아, 해당 객실에 투숙 중인 손님이 맞는 경우 숙박카드 삭제

▶ 전체 조회 기능 (100점)

- ▶ 손님이 투숙 중인 객실 번호와 손님 이름 표 형식으로 출력 (빈 방은 출력하지 말 것!)

▶ 손님 이름으로 객실 번호 조회 기능 (100점)

- ▶ 손님 이름을 입력 받아 투숙 중인 객실 번호 출력 (해당 손님이 없는 경우 안내 메시지 출력)

▶ 객실 번호로 손님 이름 조회 기능 (100점)

- ▶ 객실 번호를 입력 받아 투숙 중인 손님의 이름 출력 (빈 방인 경우 안내 메시지 출력)

▶ 메인에서는 메뉴를 출력하고, 해당 기능 선택 받아 기능을 수행하고, 완료하면 다시 메뉴 화면 출력 (100점)

▶ 결과 화면 캡처 순서

- ▶ 메뉴 출력 >> 전체조회 >> 입실 (5명 이상) >> 손님 이름으로 조회(성공/실패) >> 객실 번호로 조회(성공/실패) >> 퇴실 >> 전체조회 >> 퇴실한 번호로 조회 >> 퇴실한 손님으로 조회

▶ ls -al로 숙박 파일 정보 확인

과제 (3)

▶ 보고서 첫 장에 다음 표 작성

과제	완성 여부 (O, X)	비고
예제 2-1 (10)	O	
예제 2-2 (10)	O	
예제 2-3 (10)	O	
예제 2-4 (10)	O	
예제 2-5 (10)	O	
예제 2-6 (10)	O	
예제 2-7 (10)	X	
예제 2-8 (10)	X	
예제 2-9 (10)	X	
예제 2-10 (10)	X	
응용 1 (100)	O	
응용 2 (200)	O	
응용 3 (700)		투숙 기능 완성, 조회 기능 오류, 퇴실 기능 미완성, ...

▶ 제출기한 : **[연장]** 10월 3일 자정 (<http://cms.mmu.ac.kr/bear>)

과제 힌트 (1)

▶ 구조체

- ▶ 구조체의 멤버 중 문자열의 크기가 4의 배수가 아닐 때 어떤 일이 일어나는가?
- ▶ 다음 구조체의 실제 크기는 ?

```
typedef struct {  
    char    name[10];  
    int     no_of_guests;  
    char    sex; // 'm' or 'w'  
} guest_card;
```

▶ <string.h> - 문자열 관련 주요 함수

- ▶ **strncmp**
 - ▶ 2개의 문자열 비교 (동일하면 0 반환)
- ▶ **strncpy**
 - ▶ 문자열 복사
- ▶ **memset**
 - ▶ 메모리 영역을 특정 값으로 채움 (보통 0으로 초기화)
 - ▶ 구조체 초기화

과제 힌트 (2)

▶ Makefile

```
CC = gcc
CFLAGS = -g
OBJS = ext3.o show_menu.o guest_in.o check_out.o list.o search_by_name.o search_by_room_no.o

ext3: $(OBJS)
    $(CC) $(CFLAGS) -o ext3 $(OBJS)
```

▶ 임시 파일 예 (search_by_name.c)

```
void search_by_name()
{
    // Stub function -- 아직 구현하지 않은 함수
}
```

과제 힌트 (3)

▶ ext3.c (일부)

```
int end_of_program = 0;
int choice;
char garbage[255];

...

while (!end_of_program) {
    show_menu();
    printf("Choose the menu no : ");
    if (scanf("%d", &choice) == 1) {
        switch (choice) {
            case 1:
                guest_in();
                break;

                ...

            case 6:
                end_of_program = 1;
                break;
            default:
                printf("Invalid choice !\n");
        }
    }
    else {
        printf("Invalid input !\n");
        scanf("%s", garbage);
    }
}

...
```


과제 힌트 (4)

▶ guest_in.c (일부)

```
...  
  
    if (new_card.no_of_guests > 0) {  
        fd = open(DATA_FILE_NAME, O_RDWR);  
  
        if (fd > 0) { // 기존 숙박 파일 존재  
            lseek(fd, (room_no-1)*sizeof(new_card), SEEK_SET);  
            read(fd, &org_card, sizeof(org_card));  
  
            if(org_card.no_of_guests == 0) {  
                lseek(fd, (room_no-1)*sizeof(new_card), SEEK_SET);  
                write(fd, &new_card, sizeof(new_card));  
            }  
            else {  
                printf("빈 방이 아닙니다!!!!\n");  
            }  
            close(fd);  
        }  
        else { // 숙박 파일이 존재하지 않는 경우 - 새로 생성  
            fd = open(DATA_FILE_NAME, O_CREAT | O_RDWR, S_IRWXU);  
  
            if (fd > 0) {  
                lseek(fd, (room_no-1)*sizeof(new_card), SEEK_SET);  
                write(fd, &new_card, sizeof(new_card));  
                close (fd);  
            }  
        }  
    }  
  
}
```

...

과제 힌트 (5)

▶ list.c (버전 1 - 일부)

```
...  
  
int      room_no = 1;  
  
guest_card_t  g_record;  
  
memset(&g_record, 0, sizeof(g_record));  
fd = open(DATA_FILE_NAME, O_RDONLY);  
  
if (fd > 0) {  
    while(read(fd, &g_record, sizeof(g_record)) > 0) {  
        if (g_record.no_of_guests != 0) {  
            printf("%10d호실 \t이름 : %-20s 투숙 인원 : %d \n",  
                   room_no, g_record.name, g_record.no_of_guests);  
        }  
        room_no ++;  
        memset(&g_record, 0, sizeof(g_record));  
    }  
    close(fd);  
}  
  
...
```

과제 힌트 (6)

▶ list.c (버전 2 - 일부)

```
...  
  
int room_no = 1;  
off_t last_pos, cur_pos;  
guest_card_t g_record;  
  
memset(&g_record, 0, sizeof(g_record));  
fd = open(DATA_FILE_NAME, O_RDONLY);  
  
if (fd > 0) {  
    last_pos = lseek(fd, 0, SEEK_END); // file 끝 위치 저장  
  
    cur_pos = lseek(fd, 0, SEEK_SET);  
    while(cur_pos < last_pos) {  
        read(fd, &g_record, sizeof(g_record));  
        if (g_record.no_of_guests != 0) {  
            printf("%10d호실 \t이름 : %-20s 투숙 인원 : %d \n",  
                room_no, g_record.name, g_record.no_of_guests);  
        }  
        room_no ++;  
        memset(&g_record, 0, sizeof(g_record));  
        cur_pos = lseek(fd, 0, SEEK_CUR); // 현재 위치 저장  
    }  
  
    close(fd);  
}  
  
...
```

강의 내용

- ▶ 표준라이브러리를 이용한 파일 입출력
 - ▶ 강의 내용 (26/51 ~ 42/51)
 - ▶ <http://lily.mmu.ac.kr/lecture/13u2/ch02.pdf>
- ▶ 파일기술자와 파일포인터간 변환
 - ▶ 강의 내용 (43/51 ~ 45/51)
- ▶ 임시파일 사용
 - ▶ 강의 내용 (46/51 ~ 49/51)

과제 (1)

- ▶ [예제 2-11], [예제 2-12] (각 10점)
- ▶ [예제 2-13], [예제 2-14] (각 10점)
- ▶ [예제 2-15], [예제 2-16], [예제 2-17] (각 10점)
- ▶ [예제 2-18], [예제 2-19] (각 10점)
- ▶ [예제 2-20], [예제 2-21] (각 10점)
- ▶ 응용 #4 (200점)
 - ▶ 응용 #3을 표준 라이브러리를 활용한 버전으로 작성하시오.
- ▶ 응용 #5
 - ▶ 텍스트 파일에 이름, 전화번호를 저장하고, 검색하는 프로그램 작성 (500점)
 - ▶ 기록 (이름, 전화번호)
 - ▶ 전체 출력
 - ▶ 이름으로 전화번호 검색
 - ▶ 전화번호로 이름 검색
 - ▶ 메인에서 메뉴 출력 후 해당 기능 수행 - 완료되면 다시 메뉴 출력

홍길동	(010) 123-4567
이몽룡	(010) 234-5678
김하나	(010) 345-6789

과제 (2)

▶ 보고서 첫 장에 다음 표 작성

과제	완성 여부 (O, X)	비고
예제 2-11 (10)	O	
예제 2-12 (10)	O	
예제 2-13 (10)	O	
예제 2-14 (10)	O	
예제 2-15 (10)	O	
예제 2-16 (10)	O	
예제 2-17 (10)	X	
예제 2-18 (10)	X	
예제 2-19 (10)	X	
예제 2-20 (10)	X	
예제 2-21 (10)	X	
응용 4 (200)		투숙 기능 완성, 조회 기능 오류, 퇴실 기능 미완성, ...
응용 5 (500)		저장 완성, 전체 출력 완성, 이름 검색 완성, 번호 검색 미완성, ...

▶ 제출기한 : 10월 7일 자정 (<http://cms.mmu.ac.kr/bear>)

과제 힌트 (1)

▶ Makefile

```
CC = gcc
CFLAGS = -g
OBJS = ext5.o show_menu.o add_entry.o list.o search_by_name.o search_by_phone_no.o

ext3: $(OBJS)
    $(CC) $(CFLAGS) -o ext5 $(OBJS)
```

▶ 임시 파일 예 (search_by_name.c)

```
void search_by_name()
{
    // Stub function -- 아직 구현하지 않은 함수
}
```

과제 힌트 (2)

▶ add_entry.c (버전 1 - 일부)

```
...  
    fgets(line, LINE_LENGTH, stdin);           // 버퍼에 있는 CR-LF 처리 - 앞선 scanf의 흔적  
  
    printf("추가하고자 하는 이름과 전화번호를 입력해주세요 : ");  
    fgets(line, LINE_LENGTH, stdin);  
  
    if (sscanf(line, "%s %s", tmp1, tmp2) == 2) { // 문자열이 2개인지 확인  
        ofp = fopen(DATA_FILE_NAME, "a");  
        if (ofp) {  
            fputs(line, ofp);  
            fclose(ofp);  
        }  
        else {  
            perror(DATA_FILE_NAME);  
        }  
    }  
    else {  
        printf("잘못된 입력입니다.\n");  
    }  
...  

```


과제 힌트 (3)

▶ add_entry.c (버전 2 - 일부)

```
...  
    printf("추가하고자 하는 이름을 입력해주세요 : ");  
    scanf("%s", name);  
    printf("추가하고자 하는 전화번호를 입력해주세요 : ");  
    scanf("%s", phone_no);  
  
    sprintf(line, "%s %s\n", name, phone_no);  
  
    ofp = fopen(DATA_FILE_NAME, "a");  
    if (ofp) {  
        fputs(line, ofp);  
        fclose(ofp);  
    }  
    else {  
        perror(DATA_FILE_NAME);  
    }  
}
```

...

과제 힌트 (4)

▶ list.c (일부)

```
...  
  
    ifp = fopen(DATA_FILE_NAME, "r");  
  
    if (ifp) {  
        while(fgets(line, LINE_LENGTH, ifp)) {  
            fputs(line, stdout);  
        }  
        fclose(ifp);  
    }  
    else {  
        perror(DATA_FILE_NAME);  
    }  
  
...
```