

# C프로그래밍 언어 실습 14 주차

# 실습 1

## 회문 판별 프로그램

```
#include <stdio.h>

//문자열이 회문인지 아닌지를 알려주는 함수
void pal(char* ptr, int len){
    int count = 0;
    for (int i = 0; i < len / 2; i++){
        //i번째 맨 처음과 맨 마지막 문자가 같으면 count값 증가
        if (ptr[i] == ptr[len - i - 1])
            count++;
        //같지 않으면 반복문 탈출
        else{
            printf("회문이 아닙니다.");
            break;
        }
    }
    //count값이 len/2와 같아야 회문
    if (count == len / 2)
        printf("회문입니다.");
}
```

```
//문자열의 길이를 알려주는 함수
int leng(char* ptr){
    int leng = 0;

    //null문자를 만날때까지 반복
    while (ptr[leng] != '\0')
        leng++;

    return leng;
}

int main(void){
    char str[100];
    int length;
    printf("문자열 입력 : ");
    scanf_s("%s", str, 100);
    length = leng(str); //문자열의 길이를 알기 위해 leng함수 호출
    pal(str, length); //문자열과 길이를 인자로 pal 함수에 전달

    printf("\n");
    return 0;
}
```

# 실습 2

파일 입출력 및 구조체 등을 이용하여 성적관리 프로그램을 작성하시오.

(다음조건 활용)

[입력양식] 홈페이지 첨부파일 input.txt 사용

Grade 조건 (A+) = 90, 90 > A0 = 80, 80 > B+ = 70, 70 > B0 = 60, 60 > F)

```
struct s{  
int num[6];  
char name[20];  
int kor;  
char grade[1];  
} stu[STUDENT];
```

C:\Windows\system32\cmd.exe

```
000006 hwang in su      99 A+
000007 hwang sun su    100 A+
000008 hwang man su    30 F
000009 hwnag duck su    100 A+
000010 kim in              78 B+
000011 moon huck ju       65 B0
000012 lee huui jki       100 A+
000013 kim jkim hn        60 B0
000014 seung yong un     100 A+
000015 kim ujik jkim     67 B0
000016 moon hjki hik    77 B+
000017 honh seyng ha     99 A+
000018 min yong sik      100 A+
000019 lee hayng ran     100 A+
000020 chu chang hua      90 A+
```

output.txt - 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

\*\*\*\*\* \* RECORD FILE \* \*\*\*\*\*

NO	NAME	KOR	GRADE
000006	hwang in su	99	A+
000007	hwang sun su	100	A+
000008	hwang man su	30	F
000009	hwnag duck su	100	A+
000010	kim in	78	B+
000011	moon huck ju	65	B0
000012	lee huui jki	100	A+
000013	kim jkim hn	60	B0
000014	seung yong un	100	A+
000015	kim ujik jkim	67	B0
000016	moon hjki hik	77	B+
000017	honh seyng ha	99	A+
000018	min yong sik	100	A+
000019	lee hayng ran	100	A+
000020	chu chang hua	90	A+
000021	kim seng min	88	A0

<http://www.parkjonghyuk.net/lecture/2018-1st-lecture/programming1/input.txt>

1

```

#include <stdio.h>
#include <string.h>
#define STUDENT 25 //학생 수
int main(){

    int i;
    struct s{
        int num[6];
        char name[20];
        int kor;
        char grade[1];
    } stu[STUDENT];

    FILE *fp;
    FILE *fp1;

    fp=fopen("input.txt","r");

    for(i=0;i<STUDENT;i++){
        fscanf(fp, "%s", &stu[i].num);
        fgets(stu[i].name, 20, fp);
        fscanf(fp, "%d", &stu[i].kor);

        if(stu[i].kor >= 90) {stu[i].grade[0] = 'A'; stu[i].grade[1]='+';}
        else if(stu[i].kor >= 80) {stu[i].grade[0] = 'A'; stu[i].grade[1]='0';}
        else if(stu[i].kor >= 70) {stu[i].grade[0] = 'B'; stu[i].grade[1]='+';}
        else if(stu[i].kor >= 60) {stu[i].grade[0] = 'B'; stu[i].grade[1]='0';}
        else {stu[i].grade[0] = 'F'; stu[i].grade[1]=0;}

```

```

        printf("%s", stu[i].num);
        printf("%s", stu[i].name);
        printf("%3d ", stu[i].kor);
        printf("%c%c ", stu[i].grade[0],stu[i].grade[1]);
        printf("\n");
    }
    fclose(fp);

    fp1 = fopen("output.txt","w");

    fprintf(fp1, "***** * RECORD FILE * *****\n");
    fprintf(fp1, "=====\n");
    fprintf(fp1,"NO          NAME          KOR  GRADE \n");
    fprintf(fp1, "=====\n");

    for(i=0;i<STUDENT;i++)
    {
        fprintf(fp1,"%s %s %5d   %c%c\n", stu[i].num, stu[i].name, stu[i].kor, stu[i].grade[0],stu[i].grade[1]);
        fprintf(fp1, "-----\n");
    }
    fclose(fp1);
}

```

# 실습 3

```
#define _CRT_SECURE_NO_WARNINGS // fopen 보안 경고로 인한 컴파일 에러 방지
#include <stdio.h> // fopen, fwrite, fclose 함수가 선언된 헤더 파일
#include <string.h> // strcpy, memset 함수가 선언된 헤더 파일

#pragma pack(push, 1) // 1바이트 크기로 정렬
struct Data {
    char c1; // 1바이트
    short num1; // 2바이트
    int num2; // 4바이트
    char s1[20]; // 20바이트
};
#pragma pack(pop) // 정렬 설정을 이전 상태(기본값)로 되돌림

int main()
{
    struct Data d1;
    memset(&d1, 0, sizeof(d1)); // 구조체 변수의 내용을 0으로 초기화

    d1.c1 = 'a'; // 문자 저장
    d1.num1 = 32100; // 2바이트 크기의 숫자 저장
    d1.num2 = 2100000100; // 4바이트 크기의 숫자 저장
    strcpy(d1.s1, "Hello, world!"); // 문자열 저장

    FILE *fp = fopen("data2.bin", "wb"); // 파일을 쓰기/바이너리 모드(wb)로 열기
    fwrite(&d1, sizeof(d1), 1, fp); // 구조체의 내용을 파일에 저장

    fclose(fp); // 파일 포인터 닫기

    return 0;
}
```

## 파일에 구조체 쓰기

# 실습 3

## 파일에 구조체 읽기

```
#define _CRT_SECURE_NO_WARNINGS // fopen 보안 경고로 인한 컴파일 에러 방지
#include <stdio.h> // fopen, fread, fclose 함수가 선언된 헤더 파일

#pragma pack(push, 1) // 1바이트 크기로 정렬
struct Data {
    char c1; // 1바이트
    short num1; // 2바이트
    int num2; // 4바이트
    char s1[20]; // 20바이트
};
#pragma pack(pop) // 정렬 설정을 이전 상태(기본값)로 되돌림

int main()
{
    struct Data d1;

    FILE *fp = fopen("data2.bin", "rb"); // 파일을 읽기/바이너리 모드(rb)로 열기
    fread(&d1, sizeof(d1), 1, fp); // 파일의 내용을 읽어서 구조체 변수에 저장
    printf("%c %d %d %s\n", d1.c1, d1.num1, d1.num2, d1.s1); // a 32100 2100000100 Hello, world!
    fclose(fp); // 파일 포인터 닫기

    return 0;
}
```

# Q & A