

22장 그조체

## 차례

1. 구조체란 무엇인가
2. 구조체와 배열 그리고 포인터



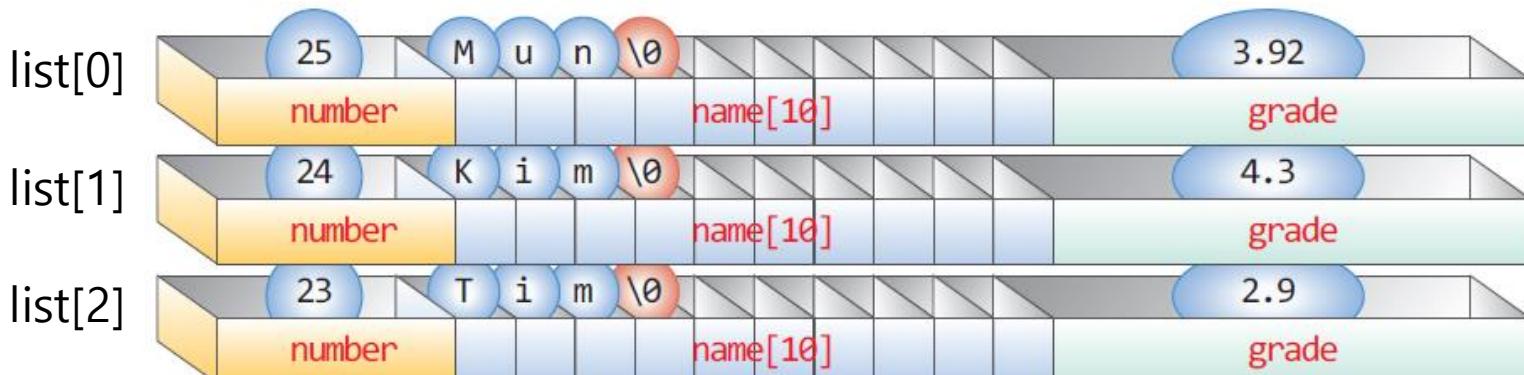
# 구조체 배열

3

- 구조체 변수를 여러 개 모아 놓은 것

```
struct student {  
    int number;  
    char name[20];  
    double grade;  
};
```

```
struct student list[3]; // 구조체 배열 선언
```





- 구조체 배열의 선언 및 참조 -> 일반적인 배열선언 및 참조방법과 동일

```
struct student {                                // 구조체 정의
    int number;
    char name[20];
    double grade;
};

struct student list[100];                      // 구조체의 배열 선언

list[0].number = 27;                            // 첫번째 배열요소에 값대입
strcpy(list[0].name, "홍길동");
list[0].grade = 178.0;
```

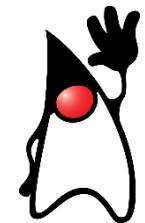


# 예제 1

5

```
struct student {  
    int number;  
    char name[20];  
    double grade;  
};  
int main(void)  
{  
    struct student list[3];  
    int i;  
    for(i = 0; i < 3; i++)  
    {  
        printf("학번을 입력하시오: ");  
        scanf("%d", &list[i].number);  
        printf("이름을 입력하시오: ");  
        scanf("%s", list[i].name);  
        printf("학점을 입력하시오(실수): ");  
        scanf("%lf", &list[i].grade);  
    }  
}
```

```
학번을 입력하시오: 230001  
이름을 입력하시오: 홍길동  
학점을 입력하시오(실수): 4.3  
학번을 입력하시오: 230002  
이름을 입력하시오: 임꺽정  
학점을 입력하시오(실수): 3.92  
학번을 입력하시오: 230003  
이름을 입력하시오: 장길산  
학점을 입력하시오(실수): 2.87  
학번: 230001, 이름: 홍길동, 학점: 4.30  
학번: 230002, 이름: 임꺽정, 학점: 3.92  
학번: 230003, 이름: 장길산, 학점: 2.87
```



## 예제 1

6

```
for(i = 0; i< 3; i++)
    printf("학번: %d, 이름: %s, 학점: %lf\n",
           list[i].number, list[i].name, list[i].grade);
return 0;
}
```

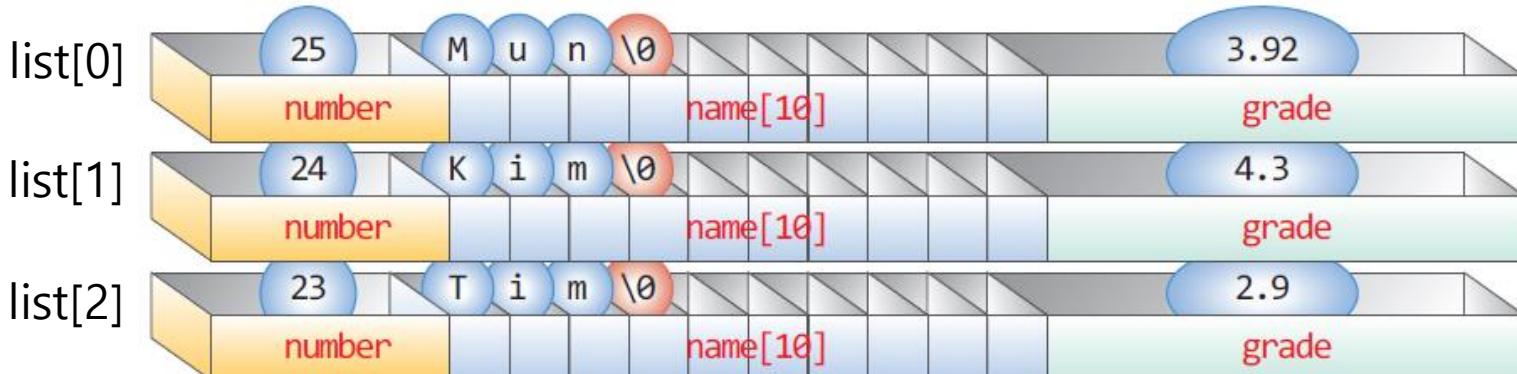


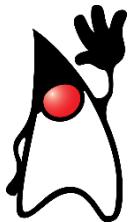
# 구조체 배열의 초기화

7

- 구조체 배열의 초기화 -> 2차원 배열의 초기화와 비슷

```
struct student list[3] = {  
    { 25, "Mun", 3.92},  
    { 24, "Kim", 4.3},  
    { 23, "Tim", 2.9}  
};
```





## 예제 2

8

```
#include <stdio.h>
struct person
{
    char name[20];
    char phoneNum[20];
    int age;
};
int main(void)
{
    struct person arr[3] = {
        {"이승기", "010-1212-0001", 21},
        {"정지영", "010-1313-0002", 22},
        {"한지수", "010-1717-0003", 19}
    };
    int i;
    for (i = 0; i < 3; i++)
        printf("%s %s %d\n", arr[i].name, arr[i].phoneNum, arr[i].age);
    return 0;
}
```

이승기	010-1212-0001	21
정지영	010-1313-0002	22
한지수	010-1717-0003	19



# 구조체 포인터

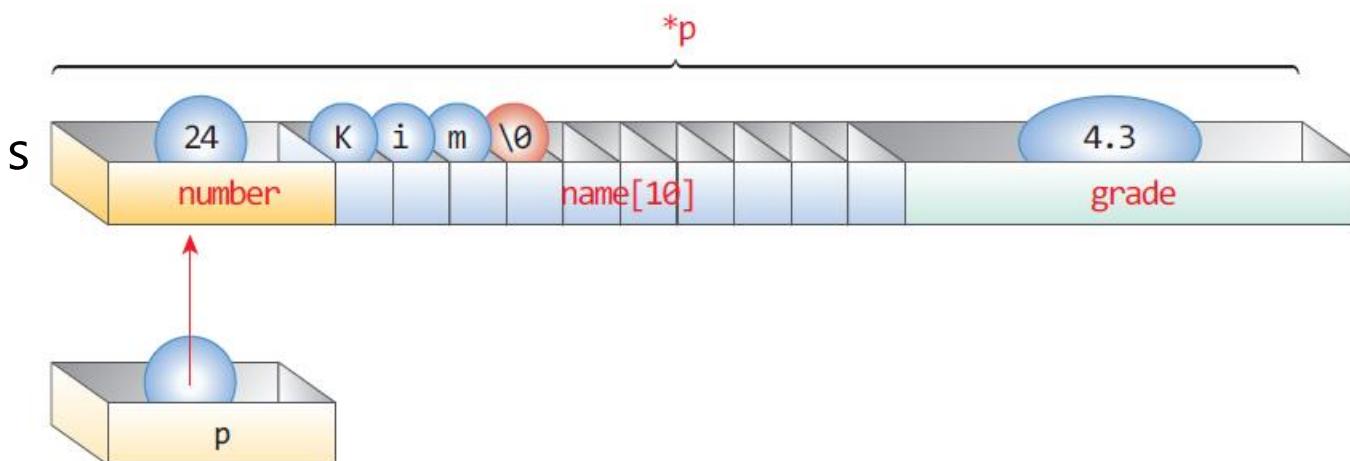
9

- 구조체 포인터 : 구조체 변수의 주소를 저장하는 변수

```
struct student s = { 24, "Kim", 4.3 };
```

```
struct student* p;
```

```
p = &s; // 구조체 전체(22바이트)를 가리키는 주소
```



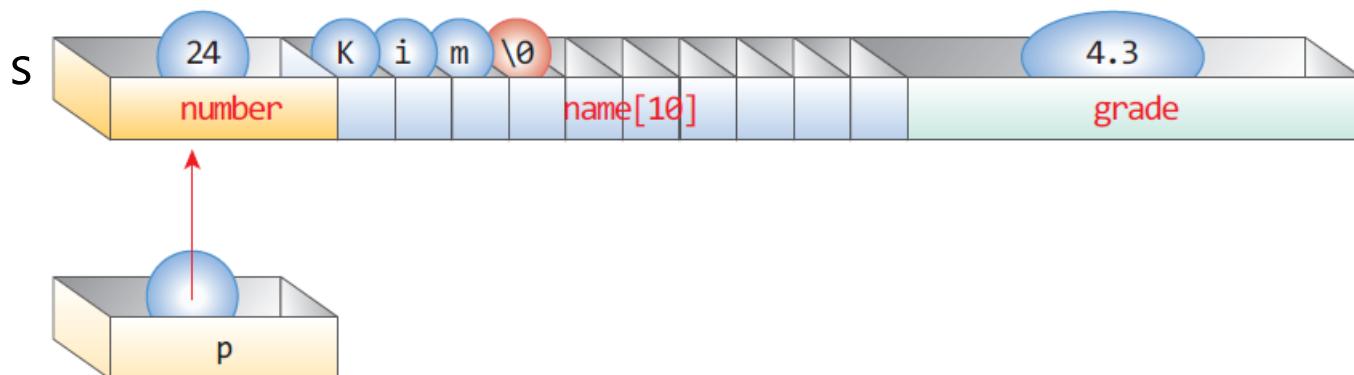


# 구조체 포인터

10

- 구조체변수의 주소는 구조체 변수의 첫번째 멤버변수의 주소와 동일

```
&s => &s.number // 멤버변수가 정의한 순서대로 할당됨
```





## 예제 3

11

```
#include <stdio.h>
struct point
{
    int xpos;
    int ypos;
};
struct person
{
    char name[20];
    char phoneNum[20];
    int age;
};
int main(void)
{
    struct point pos = { 10, 20 };
    struct person man = { "이승기", "010-1212-0001", 21 };

    printf("%p %p \n", &pos, &pos.xpos);
    printf("%p %p \n", &man, man.name);
    return 0;
}
```

```
0000006408AFF8A8 0000006408AFF8A8
0000006408AFF8C8 0000006408AFF8C8
```



## -> 연산자

12

- 간접멤버연산자 -> : 구조체 포인터로 멤버를 참조할 때 사용

// 직접멤버 참조

s.number

s.name

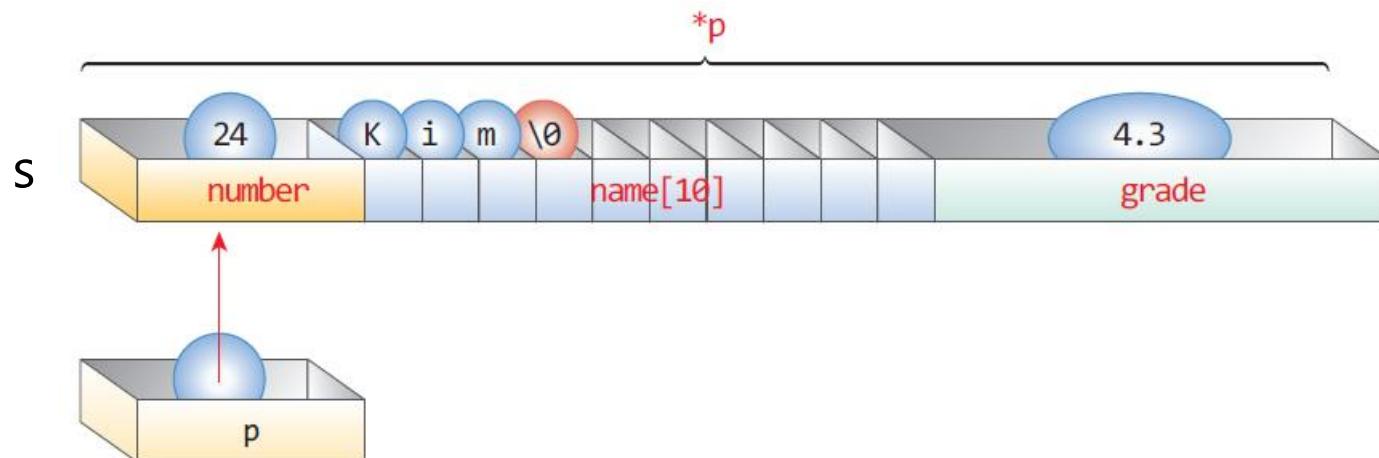
s.grade

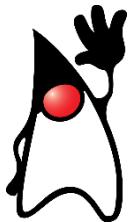
//구조체포인터를 이용한 간접멤버 참조

p->number => (\*p).number

p->name => (\*p).name

p->grade => (\*p).grade





## -> 연산자

13

- 간접멤버연산자 -> : 구조체 포인터로 멤버를 참조할 때 사용

```
struct student s = { 24, "Kim", 4.3 };
```

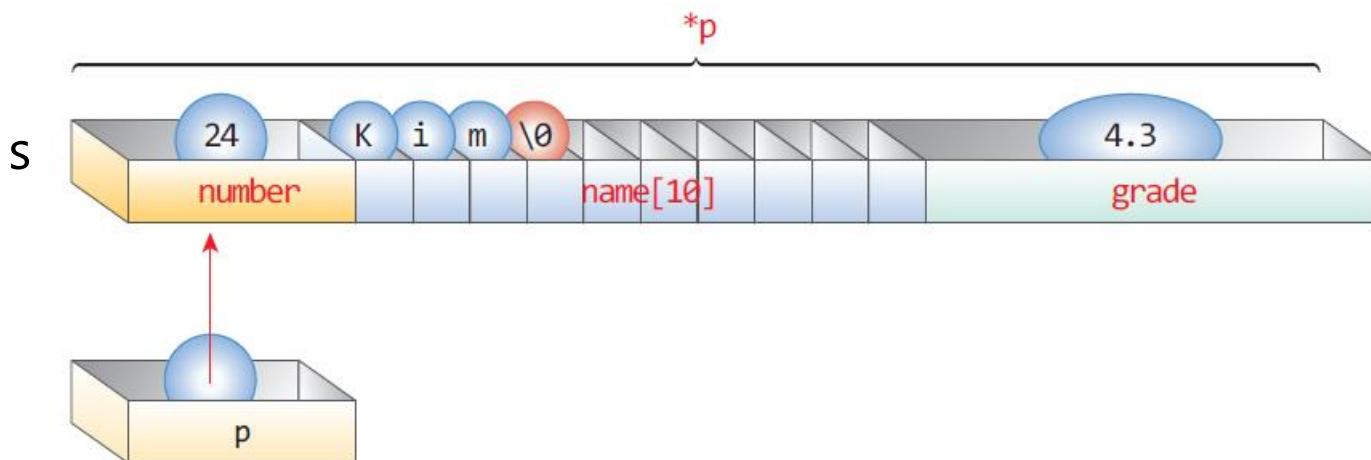
```
struct student* p;
```

```
p = &s;
```

```
printf("학번:%d 이름:%s 학점:%lf \n", s.number, s.name, s.grade);
```

```
printf("학번:%d 이름:%s 학점:%lf \n", p->number, p->name, p->grade);
```

```
printf("학번:%d 이름:%s 학점:%lf \n", (*p).number, (*p).name, (*p).grade);
```





## 예제 4

14

```
#include <stdio.h>
struct student {
    int number;
    char name[20];
    double grade;
};
int main(void)
{
    struct student s = { 20070001, "홍길동", 4.3};
    struct student* p;
    p = &s;
    printf("학번=%d 이름=%s 키=%f\n", s.number, s.name, s.grade);
    printf("학번=%d 이름=%s 키=%f\n", p->number, p->name, p->grade);
    printf("학번=%d 이름=%s 키=%f\n", (*p).number, (*p).name, (*p).grade);

    return 0;
}
```

학번=20070001 이름=홍길동 학점=4.300000  
학번=20070001 이름=홍길동 학점=4.300000  
학번=20070001 이름=홍길동 학점=4.300000



# 구조체를 멤버로 가지는 구조체

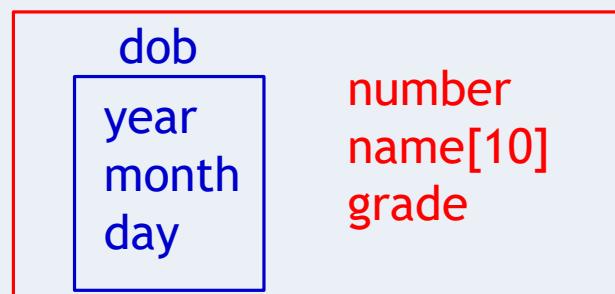
15

- 구조체의 멤버 변수는 모든 자료형이 가능 -> 구조체도 자료형이므로 구조체 멤버에 구조체 변수가 올 수 있음

```
struct date {  
    int year;  
    int month;  
    int day;  
};  
  
//구조체 멤버에 구조체 변수선언  
struct student {  
    int number;  
    char name[10];  
    struct date dob;  
    double grade;  
};
```

```
// 구조체 변수 선언  
struct student s1;  
// 멤버 참조  
s1.dob.year = 1983;  
s1.dob.month = 03;  
s1.dob.day = 29;
```

s1





## 예제 5

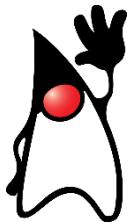
16

```
#include <stdio.h>
struct point {
    int x;
    int y;
};
struct rect {
    struct point p1;
    struct point p2;
};
int main(void)
{
    struct rect r;
    int w, h, area, peri;
```

왼쪽 상단의 좌표를 입력하시오: 1 1<엔터>  
오른쪽 하단의 좌표를 입력하시오: 6 6 <엔터>  
면적은 25이고 둘레는 20입니다.

//왼쪽상단 좌표  
//오른쪽하단 좌표





## 예제 5

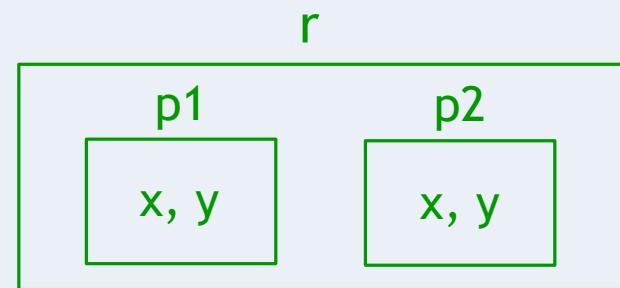
17

```
printf("왼쪽 상단의 좌표를 입력하시오: ");
scanf("%d %d", &r.p1.x, &r.p1.y);
printf("오른쪽 하단의 좌표를 입력하시오: ");
scanf("%d %d", &r.p2.x, &r.p2.y);

w = r.p2.x - r.p1.x;
h = r.p2.y - r.p1.y;
area = w * h;
peri = 2 * w + 2 * h;

printf("면적은 %d이고 둘레는 %d입니다.\n", area, peri);

return 0;
}
```





# 실습과제 1

18

- 이름, 점수를 멤버로 가지는 구조체를 정의하고 다음처럼 3명의 정보를 키보드로부터 입력 받아 최우수 학생을 구하여 화면에 출력하는 프로그램을 작성하라.
- 힌트 : 구조체배열을 사용하고 점수의 최대값을 구하는 알고리즘을 이용

이름을 입력하시오: 임꺽정<엔터>

성적을 입력하시오: 75 <엔터>

이름을 입력하시오: 홍길동 <엔터>

성적을 입력하시오: 85 <엔터>

이름을 입력하시오: 장길산 <엔터>

성적을 입력하시오: 65 <엔터>

최우수학생의 이름은 홍길동, 점수는 85점



# 실습과제 2

19

- 이름, 점수를 멤버로 가지는 구조체를 정의하고 다음처럼 3명의 정보를 키보드로부터 입력 받아 이름이 사전순서로 제일 앞에 나오는 학생을 구하여 화면에 출력하는 프로그램을 작성하라.
- 힌트 : 이름의 최소값 구하는 알고리즘을 이용

이름을 입력하시오: 임꺽정<엔터>

성적을 입력하시오: 75 <엔터>

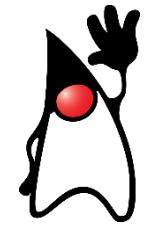
이름을 입력하시오: 홍길동 <엔터>

성적을 입력하시오: 85 <엔터>

이름을 입력하시오: 장길산 <엔터>

성적을 입력하시오: 65 <엔터>

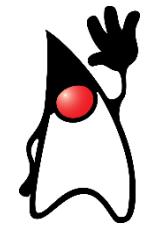
이름은 임꺽정, 점수는 75점



## 실습과제 3

20

- 실습과제 2번에서 데이터 처리부분을 구조체 포인터를 이용하여 구현해보라. 즉, 배열표현을 이용한 문장을 포인터를 이용한 문장으로 바꾸어 보라는 의미임



# 실습과제 4

21

- 교재 463페이지 문제22-2번을 변형하여 푸시오. 같은 문제를 풀면 0점처리 함



# 과제제출방법

22

- 소스코드, 라인단위의 주석, 실행결과를 포함하는 pdf파일을 작성한 후 eclass 과제 게시판에 업로드, **반드시 하나의 pdf파일로 업로드할 것**
- 기한 : 과제 게시판에 마감시간 참조
- 실행결과를 캡쳐할 때 글자를 알아보기 쉽게 확대해서 캡쳐할 것.
- 소스코드의 첫 부분은 아래처럼 제목, 날짜, 작성자(학번, 이름)를 작성할 것

```
// *****
// 제 목 : 정수 4개의 평균을 구하는 프로그램
// 날 짜 : 2023년 9월 10일
// 작성자 : 15010101 홍길동
// *****

// 소스코드 작성
```