

# **8장 다양한 자료형 하나로 묶기 : 구조체**

1. 구조체

2. 구조체 활용, 공용체, 열거형

3. 마무리

4. 프로젝트: 총점 순으로 정렬하여 출력



# 1. 이 장에서 만드는 프로그램

성적처리 프로그램



## 2. 구조체란

### (1) 구조체 정의하기

- 구조체(structure)

- 서로 관련 있는 여러 자료형 변수들을 묶어 만든 새로운 자료형
- 정보를 하나로 관리

형식

```
struct 구조체명 {  
    자료형1 변수명1;  
    자료형2 변수명2;  
    자료형3 변수명3;  
    ...  
};
```

```
struct student{ // 구조체 정의  
    int num;      // 학번(구조체 멤버)  
    double grade; // 학점(구조체 멤버)  
};
```

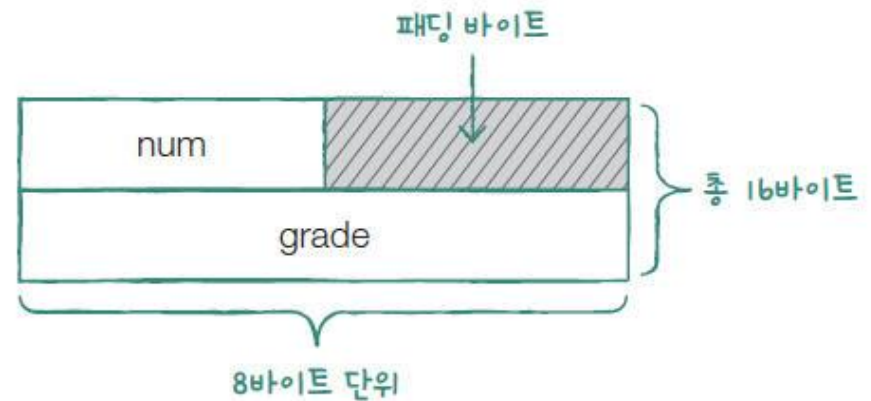


## 2. 구조체란

### (2) 구조체 변수 선언하기

형식 `struct` 구조체명 변수명;

```
struct student
{
    int num;
    double grade;
};
```



```
struct student s1;
```

```
struct student s1 = {2, 4.4};
```



## 2. 구조체란

### (2) 구조체 변수 멤버

형식    변수명.멤버명 = 값;

```
int num = 1;
```

자료형

```
struct student s1 ;
```

변수명

```
s1.num = 2;
```

```
s1.grade = 4.4;
```

```
printf("학번 : %d\n학점 : %.21f\n", s1.num, s1.grade);
```

```
struct student s1 = {2, 4.4};
```





## 2. 구조체란

### (4) 구조체 변수를 함수의 매개변수로 사용(1)

구조체를 반환하여 두 변수의 값 교환

```
01 #include <stdio.h>
02
03 struct vision
04 {
05     double left;
06     double right;
07 };
08
09 struct vision exchange (struct vision robot);
10
11 int main(void)
12 {
13     struct vision robot;
14
15     printf("시력 입력 : ");
16     scanf("%lf%lf", &(robot.left), &(robot.right));
17     robot = exchange(robot);
18     printf("바뀐 시력 : %.1lf %.1lf\n", robot.left, robot.right);
19
20     return 0;
21 }
22
```

매개변수와 반환형 모두 구조체

실행결과

시력 입력 : 15.5 20.0

바뀐 시력 : 20.0 15.5



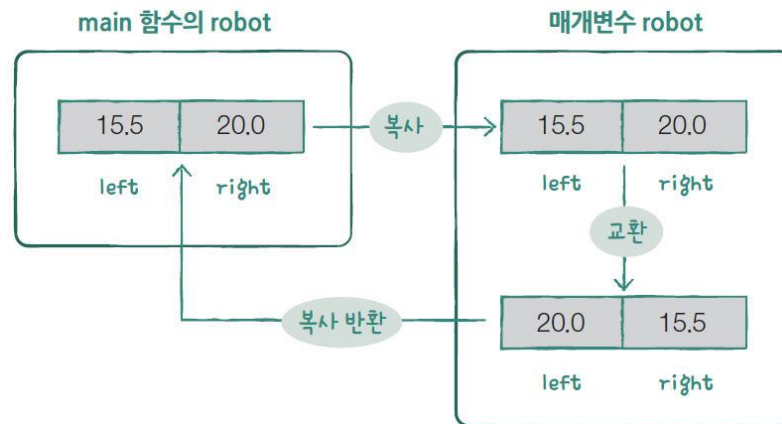


## 2. 구조체란

### (4) 구조체 변수를 함수의 매개변수로 사용(2)

구조체를 반환하여 두 변수의 값 교환

```
23 struct vision exchange(struct vision robot)
24 {
25     double temp;
26
27     temp = robot.left;
28     robot.left = robot.right;
29     robot.right = temp;
30
31     return robot;
32 }
```





# 3. 구조체 사용하기

## (1) 구조체 포인터 사용하기

- 포인터 변수 : 어떤 메모리 공간의 주소를 저장해 해당 주소를 가리키는 변수
- 구조체 포인터 : 구조체 변수의 주소를 가리키는 변수

형식    `struct` 구조체명 \* 변수명;

```
struct GameInfo * gamePtr; // 구조체 포인터 변수 선언  
gamePtr = &gameInfo1;
```



# 3. 구조체 사용

## (1) 구조체 포인터 연산자(->)

gameInfo1

사이버C	2024	33000	좋은회사
0x0001	0x0002	0x0003	0x0004

gamePtr

0x0001

```
struct GameInfo * gamePtr;  
gamePtr = &gameInfo1;
```

```
printf("\n게임(gameInfo) 출시 정보\n");  
printf("게임 이름 : %s\n", gameInfo1.name);  
printf("발매 연도 : %d\n", gameInfo1.year);  
printf("게임 가격 : %d\n", gameInfo1.price);  
printf("제작 회사 : %s\n", gameInfo1.company);  
printf("\n게임(*gamePtr) 출시 정보\n");  
printf("게임 이름 : %s\n", (*gamePtr).name);  
printf("발매 연도 : %d\n", (*gamePtr).year);  
printf("게임 가격 : %d\n", (*gamePtr).price);  
printf("제작 회사 : %s\n", (*gamePtr).company);  
printf("\n게임(gamePtr->) 출시 정보\n");  
printf("게임 이름 : %s\n", gamePtr->name);  
printf("발매 연도 : %d\n", gamePtr->year);  
printf("게임 가격 : %d\n", gamePtr->price);  
printf("제작 회사 : %s\n", gamePtr->company);
```



# 3. 구조체 사용하기

## (2) 구조체 안에 구조체 사용하기(자기 참조 구조체)

자기 참조 구조체로 list 만들기

```
01 #include <stdio.h>
02
03 struct list ←
04 {
05     int num;
06     struct list *next;
07 };
08
```

형태가 같음

```
09 int main(void)
10 {
11     struct list a = {10, 0}, b = {20, 0}, c = {30, 0};
12     struct list *head = &a, *current;
13
14     a.next = &b;
15     b.next = &c;
16
17     printf("head->num : %d\n", head->num);
18     printf("head->next->num : %d\n", head->next->num);
19
```





# 3. 구조체 사용하기

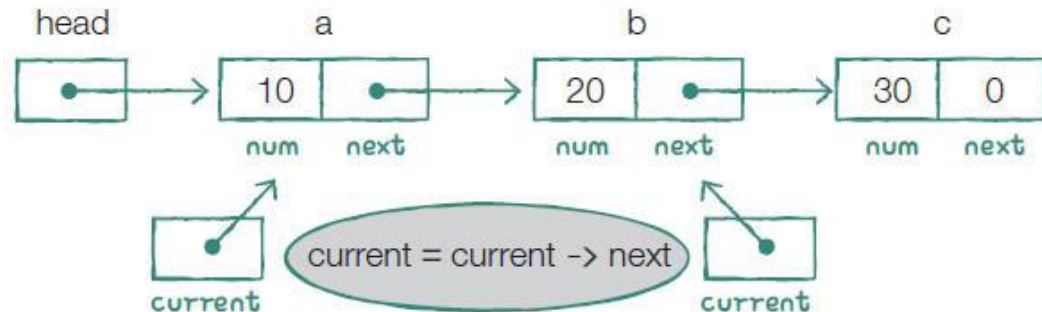
## (2) 구조체 안에 구조체 사용하기(자기 참조 구조체)

### 자기 참조 구조체로 list 만들기

```
20  printf("list all : ");
21  current = head;
22  while (current != NULL)
23  {
24      printf("%d ", current->num);
25      current = current->next;
26  }
27  printf("\n");
28
29  return 0;
30 }
```

#### 실행결과

```
head -> num : 10
head -> next-> num : 20
list all : 10 20 30
```





## 4. typedef로 구조체 선언하기

- C 언어에서 자료형에 이름을 새로 붙일 때 사용
- 자료형에 별명을 붙인다!

typedef를 사용한 자료형 재정의

```
01 #include <stdio.h>
02
03 struct student
04 {
05     int num;
06     double grade;
07 };
08 typedef struct student Student;
09 void print_data(Student *ps);
10
```

typedef struct student Student;

↓  
구조체 이름

↓  
새 자료형 이름

```
11 int main(void)
12 {
13     Student s1 = { 315, 4.2 };
14
15     print_data(&s1);
16
17     return 0;
18 }
19
20 void print_data(Student *ps)
21 {
22     printf("학번 : %d\n", ps -> num);
23     printf("학점 : %.1lf\n", ps -> grade);
24 }
```

실행결과1

학번 :	315
학점 :	4.2



## 5. 프로젝트: 성적처리 프로그램

학번 : 101  
이름 : 홍길동  
국어 영어 수학 : 90 90 95  
학번 : 102  
이름 : 윤길동  
국어 영어 수학 : 99 99 100  
학번 : 103  
이름 : 박길똥  
국어 영어 수학 : 95 75 55  
정렬전

101	홍길동	90	90	95	275	91.7
102	윤길동	99	99	100	298	99.3
103	박길똥	95	75	55	225	75.0

정렬후

102	윤길동	99	99	100	298	99.3
101	홍길동	90	90	95	275	91.7
103	박길똥	95	75	55	225	75.0

```
#include <stdio.h>
struct student
{
    int id;
    char name[20];
    int kor;
    int eng;
    int mat;
    int tot;
    double avg;
};
typedef struct student Student;
void print_data(Student * ps);
void sort(Student* arr);
int main(void){
    Student students[3];
    ...
}
```