**구조체란?**

구조체(structure type)란 사용자가 C언어의 기본 타입을 가지고 새롭게 정의할 수 있는 사용자 정의 타입.

구조체는 기본 타입만으로는 나타낼 수 없는 복잡한 데이터를 표현할 수 있다.

배열이 같은 타입의 변수 집합이라고 한다면, 구조체는 다양한 타입의 변수 집합을 하나의 타입으로 나타낸 것.

이때 구조체를 구성하는 변수를 구조체의 멤버(member) 또는 멤버 변수(member variable)라고 합니다.

#### 구조체의 정의와 선언

C언어에서 구조체는 struct 키워드를 사용하여 다음과 같이 정의함.

struct 구조체이름

{

    멤버변수1의타입 멤버변수1의이름;

    멤버변수2의타입 멤버변수2의이름;

    ...  
};

**구조체를 사용하는 상황.**

– 정보가 같이 있어야 의미가 있을 때 사용.\

Ex) (x,y좌표) (사람 이름,전화번호) (아이디, 비밀번호) 등등

**type def – struct 생략**

구조체 변수를 선언할 때, 일일이 struct 키워드를 붙이기 귀찮다면 type def을 이용하면 됨.

typedef struct 구조체이름 {

자료형 멤버이름;

} 구조체별칭;

참고 사이트 : <https://dojang.io/mod/page/view.php?id=409>

**구조체 포인터**

함수의 인자로 기본자료형 변수와 마찬가지로 구조체 변수도 포인터로 전달 가능하다. 함수 내부에서 외부에 있는 구조체 변수에 직접 접근하기 위해 구조체 포인터를 쓴다. 구조체 변수의 포인터 선언 방식과 매개변수로 구조체 포인터를 선언하는 방식은 일반적은 변수의 포인터 선언규칙을 그대로 적용하면 된다. 또한 구조체 변수의 주소 값을 얻을 때고 & 연산자를 사용한다.

**멤버로의 접근**

**(\*구조체\_포인터).멤버**

여기에서 (\*구조체\_포인터)는 구조체 변수를 의미하게 된다.

멤버에 접근하는 데 ( )가 필요한 이유는 .연산자보다 \*연산자의 우선순위가 낮기 때문이다.

그런데 포인터를 이용한 구조체 멤버의 접근을 간단히 처리하는 -> 연산자가 있는 데 다음과 같은 방식으로 접근이 가능하다.

**구조체\_포인터->멤버**

즉, "(\*구조체\_포인터).멤버"는 "구조체\_포인터->멤버"와 완전히 동일한 연산문이다.

참고 사이트 : https://neuro.tistory.com/16 [Ayoujin]

**자기참조 구조체**

struct list{

         int data;

         struct list next;

}

위 코드는 오류남. 포인터로 선언해야함.

struct list{

         int data;

         struct list \*next;  
}

위와같이 **구조체 멤버로 자기 자신의 구조체를 가르키는 포인터를 포함하는 구조체를 자기참조구조체**라고 함.

자기 참조 구조체에서 next와 같이 자기 자신의 구조체를 가르키는 포인터 변수를 링크(link)라고 함.

각 구조체는 링크를 통하여 자기 자신의 구조체를 참조하게 됨.

자기 참조 구조체는 노드 구조를 만들고 이는 **linked list, tree** 구조 등에 이용됨.

참고사이트 : <https://m.blog.naver.com/PostView.nhn?blogId=khnkd&logNo=80174067936&proxyReferer=https:%2F%2Fwww.google.com%2F>