



2019-2학기 전공 스타트 프로그램 운영 일지

교수학습개발센터
http://ctl.deu.ac.kr

학습 주제	포인터를 위한 메모리 구조, 기본 포인터 사용법 익히기		
일시	2019년 11월 21일, 17시 - 20시(3시간)	참여 학생 수	23명
장소	정보공학관 915호	강사 성명	정보공학관
학습목표			
① 포인터의 구조를 이해하고, 사용법을 익히기			
② 포인터를 위한 메모리 구조를 이해한다.			
학습 내용	<ul style="list-style-type: none"> ● 포인터 변수로 실행할 수 있는 5가지 연산 <ul style="list-style-type: none"> ① 지정(assignment) : 포인터에 번지를 지정시키는 연산, &연산자 사용 <code>int a[10], *ptr1, *ptr2; ptr1 = a; ptr2 = &a[2];</code> ② 원소 값의 획득 : * 연산자 사용 <code>x = *ptr1;</code> ③ 포인터의 메모리 번지 획득 : <code>printf("%p\n", &ptr1);</code> ④ 포인터 값의 증가와 감소 : <code>*ptr1++; *ptr2--;</code> ⑤ 두 포인터의 차 : 원소들의 상대적인 위치 파악 <code>printf("%d\n", ptr2-ptr1);</code> ● 다중 간접 참조 : 포인터가 다른 포인터를 가리키는 것으로 * 수만큼 간접 참조 일어남 <code>int i=10, *p, **q, ***r;</code> <code>p = &i, q = &p; r = &q;</code> ● 주의 사항 : C의 특성 중 하나로 프로그램 시에 매우 중요하게 사용되지만, 추적하기가 어려워 가급적 사용을 자제하는 것이 좋다. ● 1차원 배열과 포인터와의 관계 <ul style="list-style-type: none"> : 배열명은 그 배열의 첫 번째 원소의 주소(포인터) ● 배열과 포인터 변수와의 관계 <ul style="list-style-type: none"> 《 공통점 》 <ul style="list-style-type: none"> - 문자열의 초기화 가능 : <code>char a[]="Seoul", *p="Pusan";</code> - 문자열 함수에 사용 : <code>strcpy(a, "Seoul"); strcpy(p, "Pusan");</code> - 배열과 포인터 표기 서로 가능 : <code>a[1]='E'; *(a+2)='O'; *(p+1)='U'; p[2]='S';</code> 《 차이점 》 <ul style="list-style-type: none"> - 배열명은 상수(값 변경 못함), 포인터 명은 변수(값 변경 가능) - 배열명은 메모리 차지하지 않으나, 포인터 명은 메모리 차지 - 배열명은 <code>scanf()</code>의 인자가 될 수 있으나, 포인터 변수는 값이 지정된 후 가능 ● 1차원 배열의 매개변수 전달 <ul style="list-style-type: none"> ○ 실인자 : 배열은 배열 자체를 하나의 실인자로 전달할 수 없고, 배열에의 포인터를 실인자로 전달 ○ 형식인자 : ANSI C에서는 3가지 방법으로 표현 가능 ● 다차원 배열과 포인터 		

관련 사진

