프로그램의 각각의 변수 -> 메모리의 unique한 주소에 저장되어 있다.

&를 사용해서 그 변수의 주소를 가져올 수 있다. 이 ‘주소’를 pointer라 한다.

Pointer variable은 ‘주소’를 저장한다! -> memory주소에 접근할 수 있는 다른 방법!

Indirection operator \*을 사용해서 dereference pointer한다.

Array에서 array name은 array의 제일 처음을 가르키는 pointer와 같은 역할을 한다! []이 연산자는 dereference해서 보여준다.val[1] = \*(val+1)

Pointer 연산에느 ++, --, +=, -= 등으로 정수를 뺴거나 더할 수있고, pointer의 datatype에따라 8칸 이동할지, 4칸 이동할지 바뀐다.

Pointer 빼기도 되는데 결과는 정수로 몇칸이 뛰어져있는지 나온다.

Initializer -> 2가지 방법, float\* a= $integer 안됨.

Pointer는 NULL또는 0으로 초기화 가능.

Pointer에있는 address비교시 비교 연산자 사용가능.

Pointer를 function parameter로 -> reference paramter처럼 사용됨. (swap으로 두개 비교)

우리는 array를 만들 때, 필요한 만큼 초기화를 시켜줘야됨 ec) int arrr[100];

But 만약 초기화값을 모르면? -> dynamic memory allocation 동적 할당

New는 할당된 메모리의 제일 처음주소를 return해준다.

Array new 할당시

만약 memory공간이 충분하지 않으면-> 프로그램이 꺼짐. ->그러므로 delete!

Delete dbpte Delete [] arrayptr 와같이 new는 delete와 짝을 이루어야됨!

Memory leakage -> 동적할당(new) 후에 제대로 delete를 하지 않은 경우

Dangling pointer -> 메모리가 delete된 곳을 가르키고 있는 포인터.

Return Pointers from Function! -> local 변수를 return하면 안됨! Pointer만 리턴할 수있다! //만들면서!

Class 컨스트럭터에서 new 소멸자에서 delete

Class object도 동적할당 가능/constructor 파라미터는 class명 다음에 넣어준다. -> 접근시(\*daehyeon).studentID = 20161080 이렇게 해야됨

but좀더 간편하게, daehyeon->studentID = 20161080가능. 함수를 호출할때도 같은 방법!  
struct안에 있는 pointer 변수에 poinrter 객체에서 어떻게 접근해야 할까??