Queue Stack 개념 설명.

Static structure -> Array 구조

Dynamic structure -> LinkedList 구조

Queue의 경우 Head 뿐만 아니라 tail pointer를 둠으로서 for문을 사용하지 않고도 구현할 수 있다!

Console 명령어 ->shell script의 소개

(valgrind: 메모리 체크 명령어)

Git 사용? – 모든 과제를 업로드하는 것의 중요성.

OOP

* Encapsulation
* Polymorphism
* Inheritance

Lec2

Switch

Type conversion

Pass by value, pass by Reference

Overloading

Default value

Lec3

Object Oriendted! -> class

* 멤버 변수
* 멤버 함수

Private – 멤버변수 어떻게 접근?

Public

Dot operator

Constructor

Destructor

Call by value (int A) 대신 const call by reference(const int&)를 사용하는 이유?

Lec4

Struct vs Class

Union

Lec5

\*operator left, right 비교

&operator left, right 비교

Pointer 초기화 방법! NULL , 0

동적 할당New 의 메모리상의 구조!

Delete의 메모리상의 구조!

Memory learkage

Dangling pointer

Int \* ptr = new int;

Delete ptr;

Ptr ->dangling pointer! (ptr이 delete한 곳을 가르키므로 나중에 어떠한 값을 참조하게 되면 위험함 -> NULL시켜줘야됨)

Pointer return 함수 설명.

Class object pointer로 만들 때! ‘->’의 의미!

This-> 의 의미!