

특정 자료형을 위한 코딩

C++

Fundamental Data Type.

- 1. { int n; } STACK
- 2. int n; DATA
- 3. new int; Free Store

int * p = ... delete p;

sizeof(int);
sizeof n;
address of (int)

int n; keyword token

User - Defined Data Type.

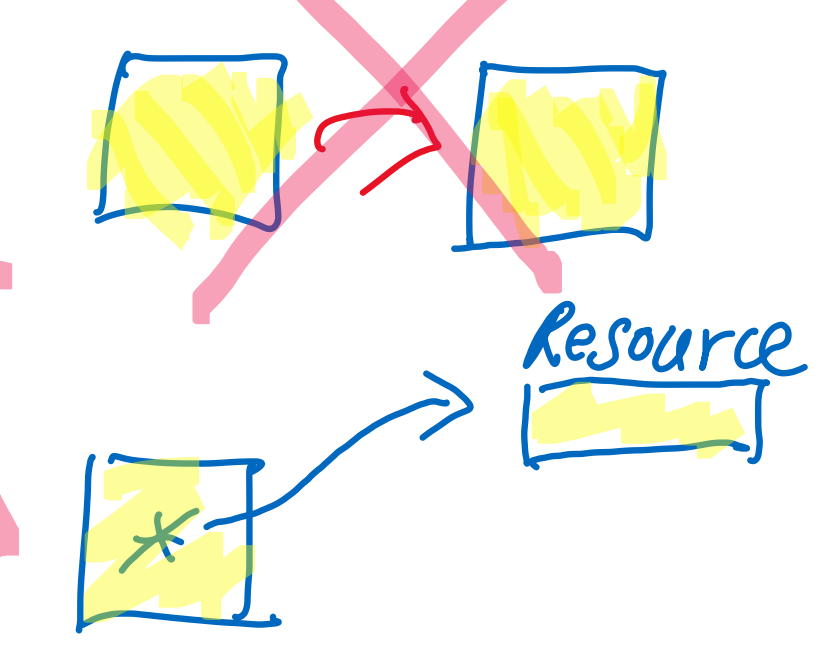
struct - 모든 member > public 일 경우.
class - default public
class - default private →

- 1. encapsulation
- 2. inheritance
- 3. polymorphism

```
class X {
private:
    variable members;
public:
    X();
    ~X();
    X(const X&);
    X& operator=(const X&);
    X(X&&);
    X& operator=(X&&);
    operator overloading
}
```

special function

生死



getter() const; ⇒ getNum(); interface fun
setter

자료형과 관계없는 코딩.

C++ keyword

Standard Template Library
Generic Programming.
함수, 클래스
↓
Algorithm Data Structure

operator overloading

lhs operand operator rhs operand

(+, -, *, /)
(<<, >>)

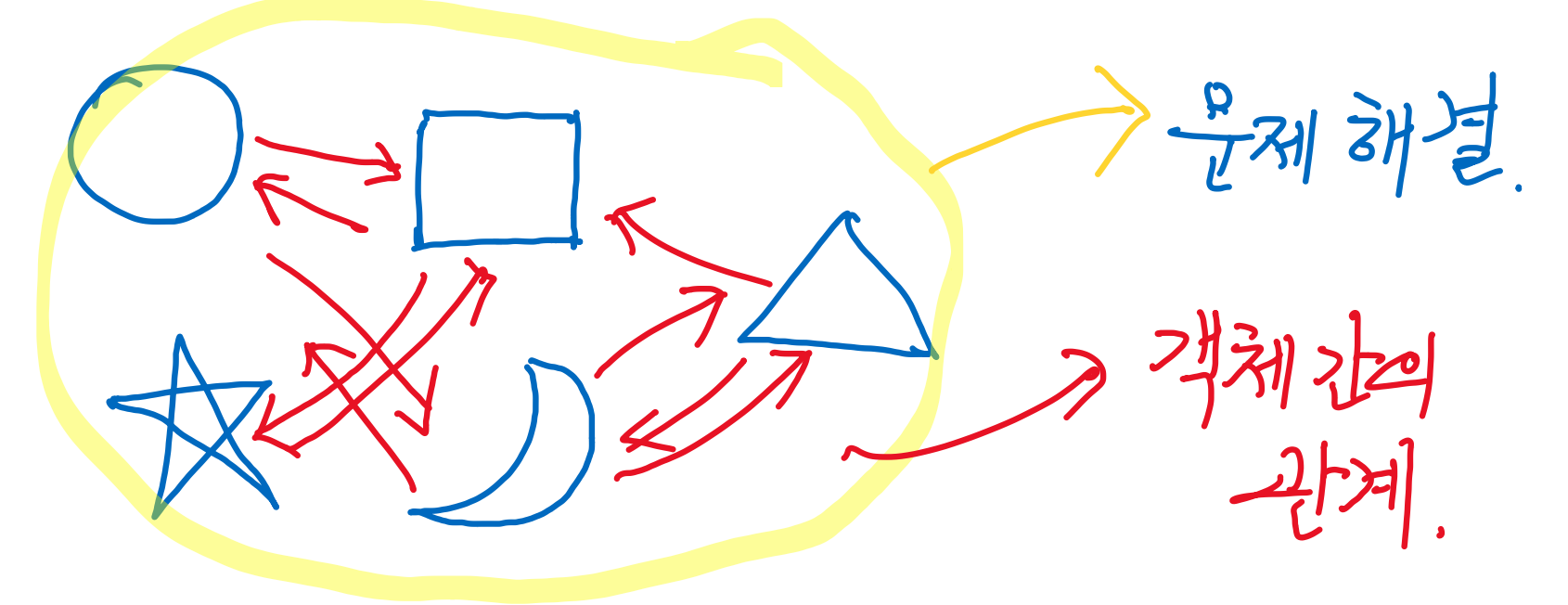
a + b

a.operator+(b);
operator+(a, b);

<<, >> friend 전역함수;

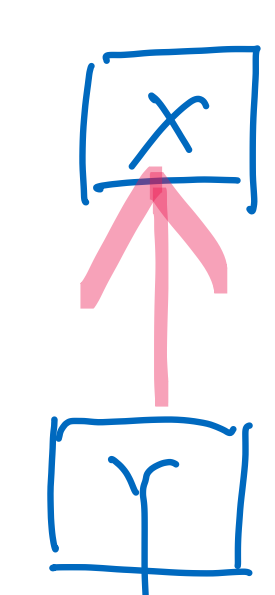
X a; a □

instantiating.



inheritance { 1. 코드 재사용
2. 다형성 구현 → virtual

class X : public Y;



모든 객체마다
vtbl을 가리키는 vptr
(8 bytes)

memory → 空間 → contiguous
time → 時間 → continuous

trade-off