

PRG2

- Repaso FUERTE Prop Estructura
 \cong 1 mes
plantamiento ex Dev } + de 1
formac

- INTRO \Rightarrow ABS/ENC/MOD/JGR

- Prop basada en objetos

- Clases y Objetos {
 - Vista pública clases
 - Vista pública objetos
 - Vista privada clases
 - Vista privada objetos

- Colaboración entre objetos

HOOD (A.K.A FEB)

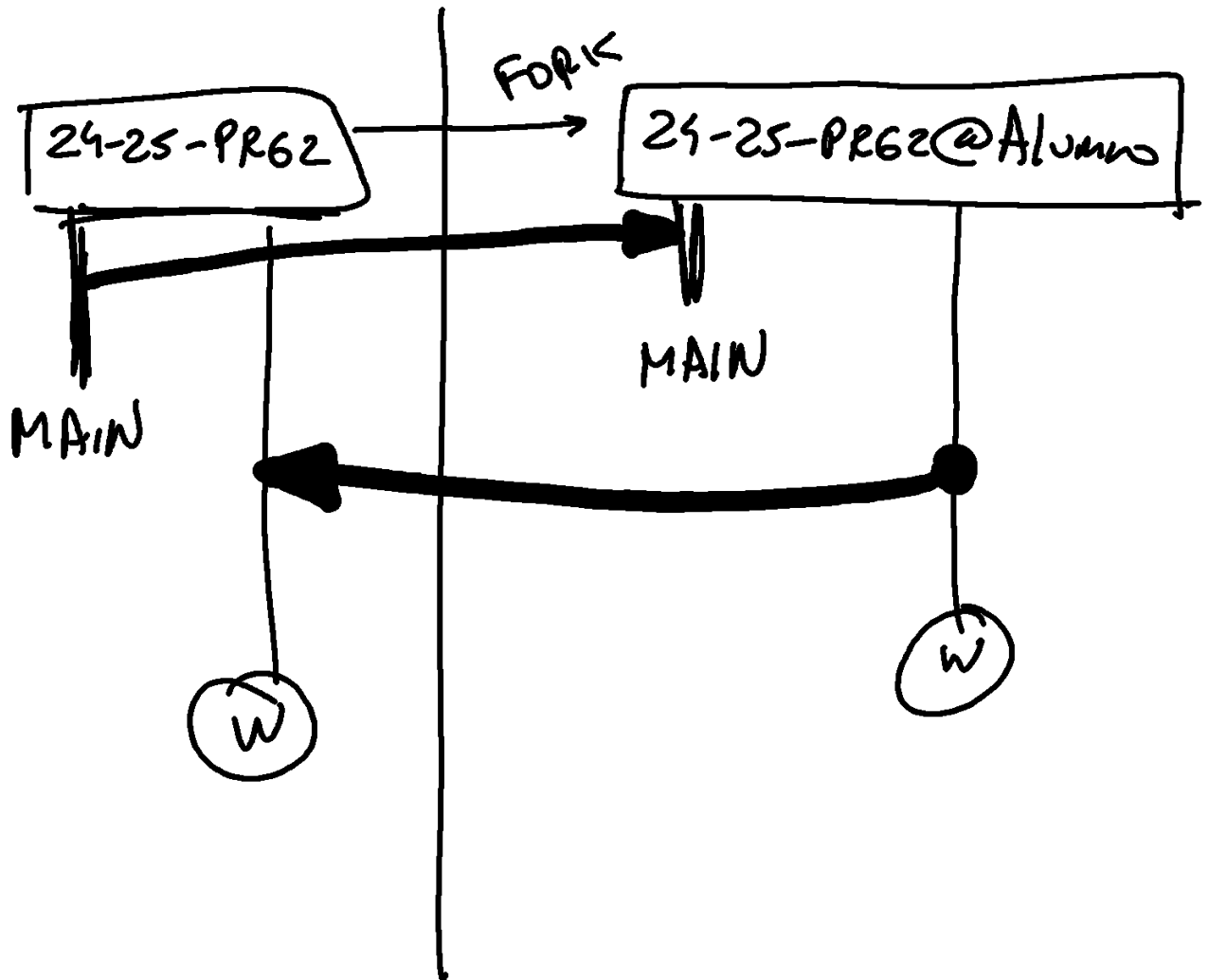
- Prop Orientada objetos

- Herencia {
 - \rightarrow x extensión
 - \rightarrow Clases abstractas
 - \rightarrow x implementación
 - \rightarrow Limitaciones y Beneficios

- Polimorfismo



Repo



PR6

Origen de los datos { Proceso, \rightarrow leng. d'orig
Datos, \rightarrow SGBD

TESIS \rightarrow Proceso
ANTITESIS \rightarrow Datos { Sintesis
POO

Nadie piense en { solo Proceso
solo Datos

OO: Cuidar datos y procesos

Bases
Prog \Rightarrow Evolució.
leng.
Prog \Rightarrow Elementos
POO \Rightarrow Objetos
POO

BASES DE LA PROGRAMACIÓN

- Abstracción → CARACTERÍSTICAS ESENCIALES IGNORANDO LO SUPERFLUO
- Encapsulación → OCULTA DETALLES DEL SOPORTE: el cómo guardo
- Modularidad → DESCOMPONER EL SISTEMA EN PIEZAS POCO ACOPLADAS Y COHESIVAS

SISTEMA → Conjunto de piezas que colaboran entre sí

POCO ACOPLADAS ↔ medida de la independencia

COHESIVAS ↔ tener significado propio, por sí mismo

- Jerárquica → ESTRUCTURACIÓN QUE PRODUCE UNA ORGANIZACIÓN JERÁRQUICA, POR COMPOSICIÓN O CLASIFICACIÓN

EVOLUCION LENG. PROGRAMACIÓN

CÓDIGO MÁQUINA

→ ENSAMBLADOR

PROG DE ALTO NIVEL }
FORTRAN
PASCAL
BASIC

↓
PROG ESTRUCTURADA

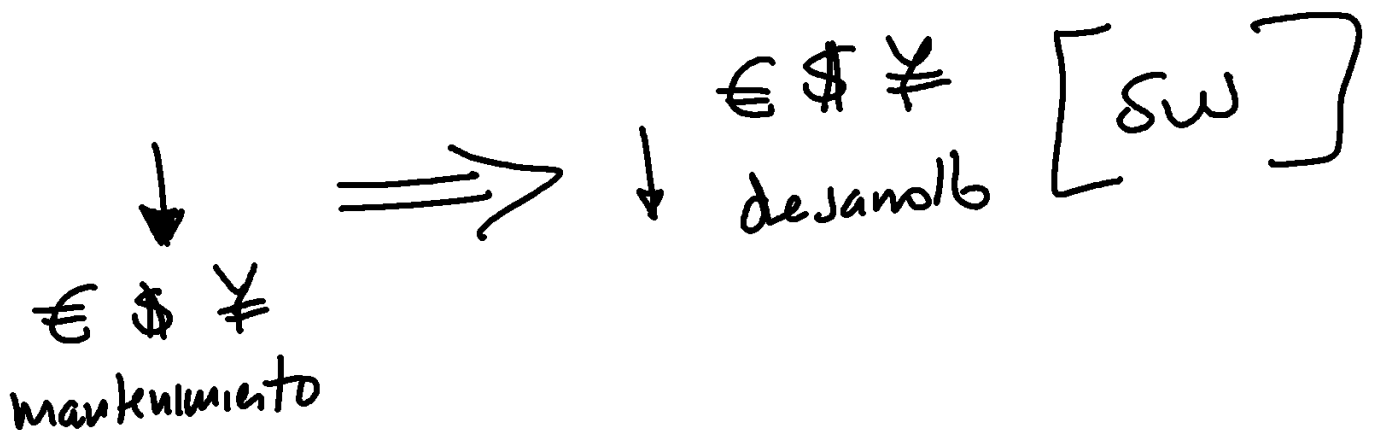
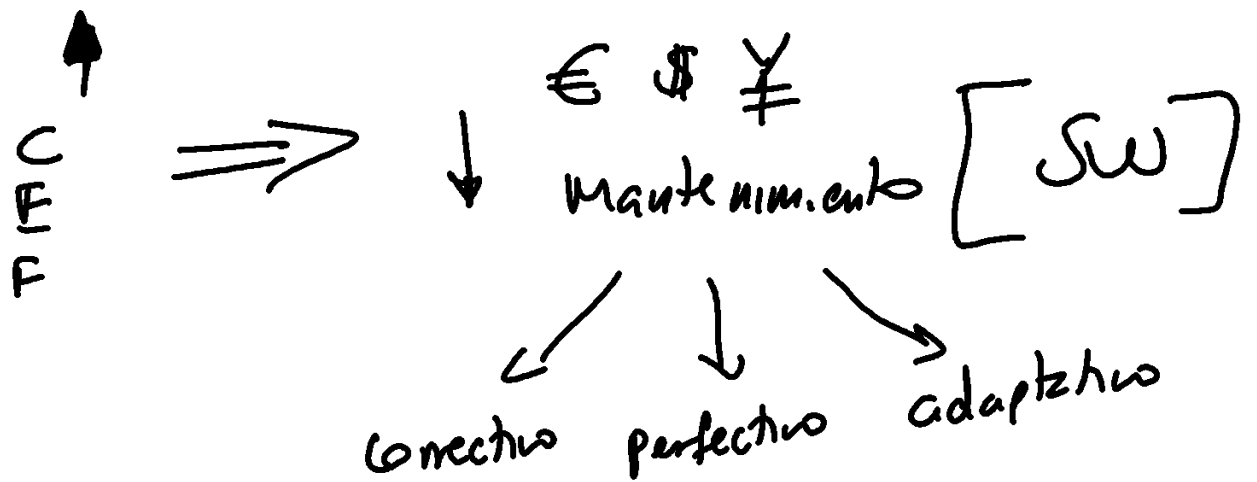
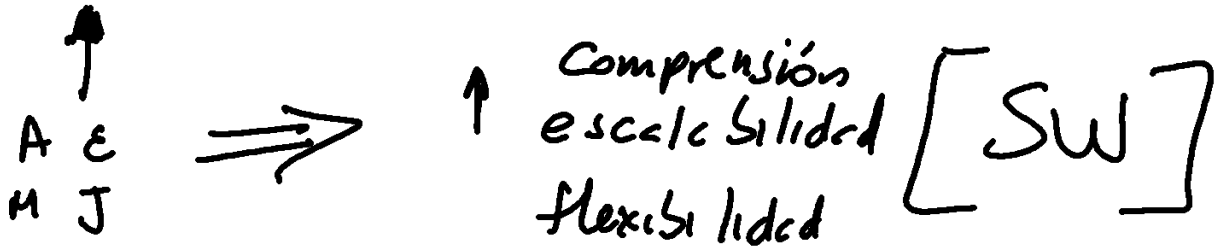
↓
PROG MODULAR

↓
TAD

↓
POO

OO = TAD + Polimorfismo

OBJETIVOS POO



LEYES DE LHEMAN Y BELADY(*)

* Ley del cambio continuo:

Ej.:
COCINA

Un programa que se usa en un ámbito del mundo real necesariamente debe cambiar o convertirse cada vez en menos útil

* Ley de complejidad creciente:

debido a que los programas cambian por evolución, su estructura se convierte en más compleja a menos que se hagan esfuerzos activos para evitar esto

(*) "deuda técnica"

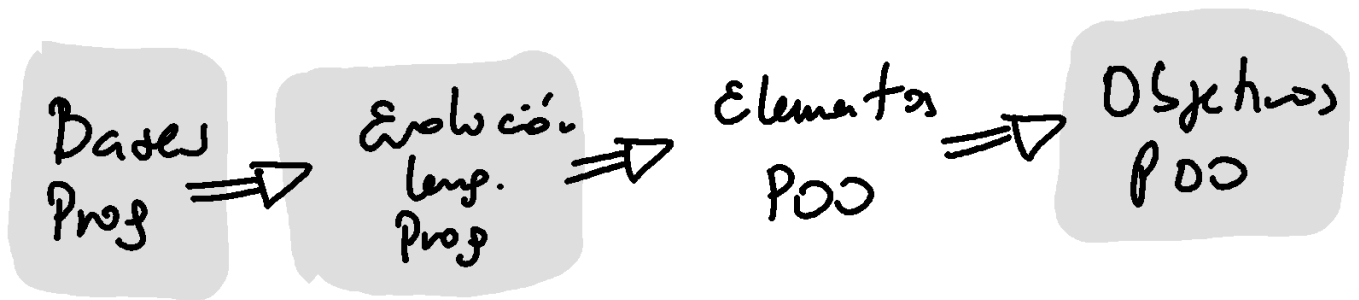
⇒ Si el código tiene que cambiarlo

Conviene que sea:

- legible

⇒ Si es legible, es mantenible

⇒ Si es mantenible, ∴ - Costes!!!



ELEMENTOS DE LA POO

CLASE { Descripción de datos y operaciones que describen el comportamiento de un conjunto de elementos homogéneos

OBJETO { Ejemplar concreto (instancia) de una clase, que responde al comportamiento definido por las operaciones de la clase a la que pertenece, adecuándose al estado de sus datos particulares

Clase

Intervalo
extremo superior extremo inferior
intersector desplazar longitud

Objeto

horas De Clase (10-12)

