

GRAU D'ENGINYERIA INFORMÀTICA  
Curs acadèmic 2018-19

# PROGRAMACIÓ II

**Problema 4:**  
**Revisió de la pràctica 1 i**  
**suport a la pràctica 2**



UNIVERSITAT DE  
BARCELONA

# Índex

- **Dubtes correccions L1**
- Preguntes memòria L2
- Diagrama de flux

# Dubtes correccions L1: equals de FitxerMultimèdia

- Podem aprofitar **l'herència**
  - De fet, s'hauria d'aprofitar: PII és sobre POO!
- Aprofitant l'herència, aquest mètode es pot implementar en dos passos:
  1. Cridant al mètode *equals* de Fitxers (super)
  2. I després comparant la descripció:
    - Per això, recordeu que cal fer una conversió de tipus
    - Recordeu també que al ser del tipus String, la comparació ha de ser amb equals, i no amb ==

# Dubtes correccions L1: equals de FitxerMultimèdia

- Podem aprofitar **l'herència**
  - De fet, s'hauria d'aprofitar: PII és sobre POO!
- Per què?
  - Un Fitxer Multimèdia és un File
  - File no té descripció

# Dubtes correccions L1: equals de FitxerMultimèdia

- **Observació:** Quan heretem un mètode, el podem sobreescrivre, però en cap cas modificar la seva signatura (això seria una sobrecàrrega)
  - Tot i que Netbeans us deixa, no és una bona pràctica de programació orientada a objectes

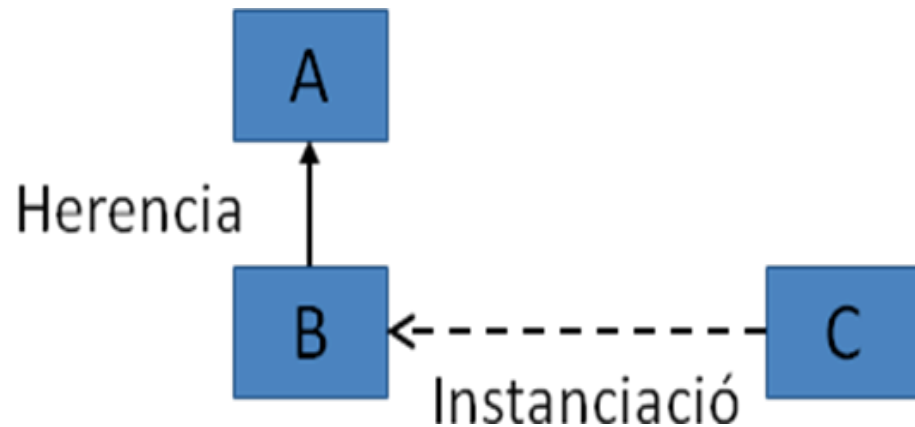
Exemple:

@override

```
public boolean equals (Object obj)
```

# Dubtes correccions L1: diagrama de classes

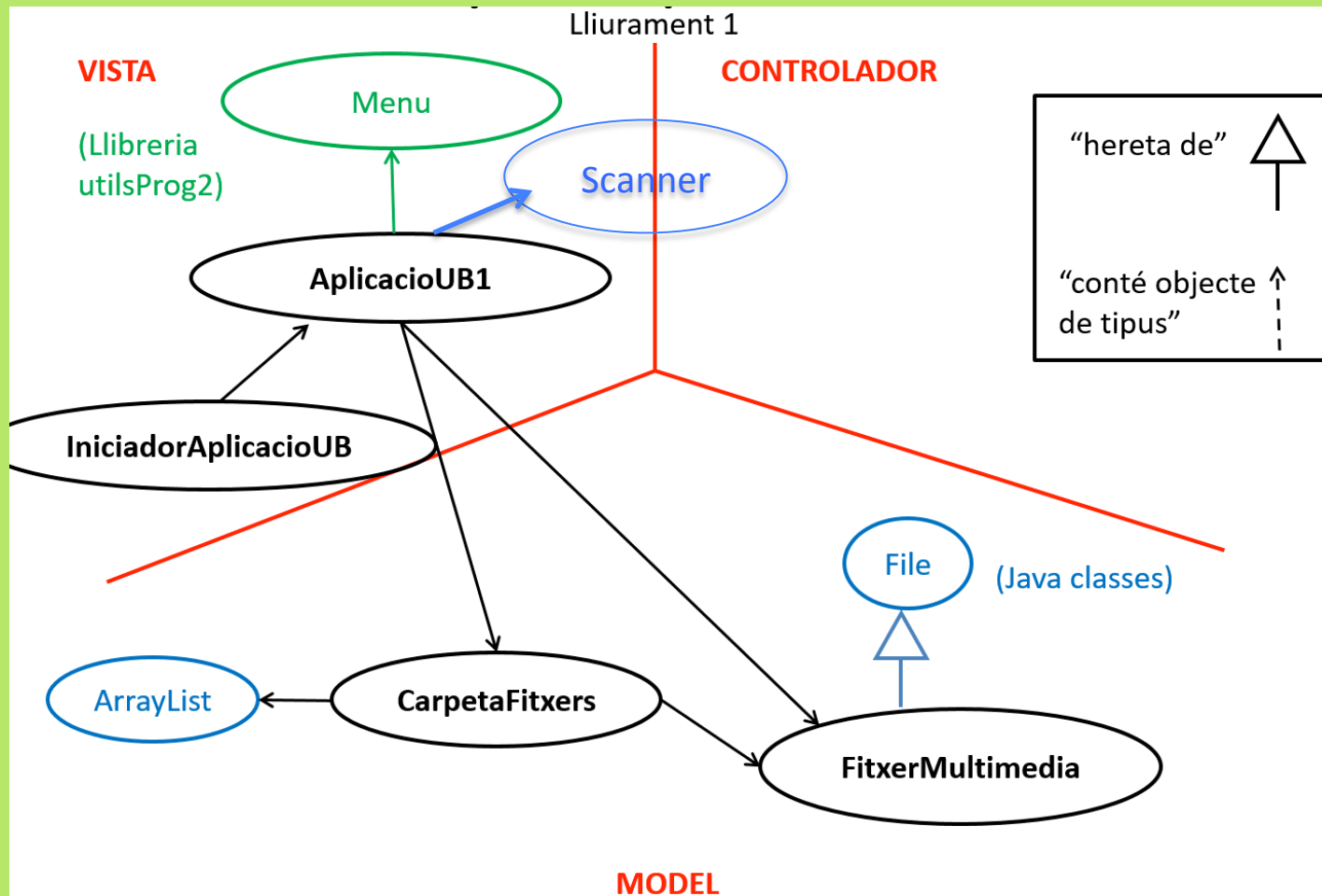
- En els diagrames, és molt important seguir les pautes indicades en els enunciats de l'assignatura, perquè segueixen (flexiblement) unes convencions que veureu més endavant en el Grau.



# Dubtes correccions L1: diagrama de classes

- Dibuixar el diagrama de relacions entre les classes que heu utilitzat a la pràctica
- Incloure tant les classes implementades com les que pertanyen a la llibreria de Java i UtilsProg2
- No cal incloure la llista de mètodes i atributs

# Dubtes correccions L1: diagrama de classes





## Dubtes correccions L1: remove

- Segons la implementació de la classe CarpetaFitxers, si tenim dos fitxers multimèdia corresponents al mateix fitxer, quan cridem al mètode per eliminar un d'aquests fitxers, eliminarà l'altre fitxer o no?

# Dubtes correccions L1: remove

- Carpeta de Fitxers té un atribut del tipus ArrayList, i aquest ja té un mètode remove
- **Opció 1.** Si heu implementat el mètode eliminar cridant al mètode remove de ArrayList, la resposta a la pregunta és NO. Veiem-ho

# Dubtes correccions L1: remove cridant a remove de ArrayList

```
public boolean ↴ remove(Object o) {  
    if (o == null) {  
        for (int index = 0; index < size; index++)  
            if (elementData[index] == null) {  
                fastRemove(index);  
                return true;  
            }  
    } else {  
        for (int index = 0; index < size; index++)  
            if (o.equals(elementData[index])) {  
                fastRemove(index);  
                return true;  
            }  
    }  
    return false;  
}
```

<http://greppcode.com/project/repository.greppcode.com/java/root/jdk/openjdk/>

# Dubtes correccions L1: remove cridant a remove de ArrayList

```
public boolean ↴ remove(Object o) {  
    if (o == null) {  
        for (int index = 0; index < size; index++)  
            if (elementData[index] == null) {  
                fastRemove(index);  
                return true;  
            }  
    } else {  
        for (int index = 0; index < size; index++)  
            if (o.equals(elementData[index])) {  
                fastRemove(index);  
                return true;  
            }  
    }  
    return false;  
}
```

A quin equals es crida a la instrucció `o.equals()`?

<http://greppcode.com/project/repository.greppcode.com/java/root/jdk/openjdk/>

# Dubtes correccions L1: remove amb un iterador

- Carpeta de Fitxers té un atribut del tipus ArrayList, i aquest ja té un mètode remove
- **Opció 2.** Si heu implementat aquest mètode amb un iterador, llavors podeu aconseguir esborrar totes les instàncies

# Índex

- Dubtes correccions L1
- **Preguntes memòria L2**
- Diagrama de flux

# Preguntes memòria L2

- En aquesta part ens centrarem en algunes de les preguntes que heu de contestar a la memòria del lliurament

# Preguntes memòria L2: # objectes creats

- Quants objectes s'han creat després de la següent instrucció?

```
AplicacióUB2 aplicació = new AplicacióUB2();
```



## • Preguntes memòria L2: # objectes creats

- Quants objectes s'han creat després de la següent instrucció?

AplicacióUB2 aplicació = new AplicacióUB2();

- Estratègia: mireu el constructor de AplicacióUB2
- AplicacióUB2 > Controlador > Dades > Reproductor...

# Índex

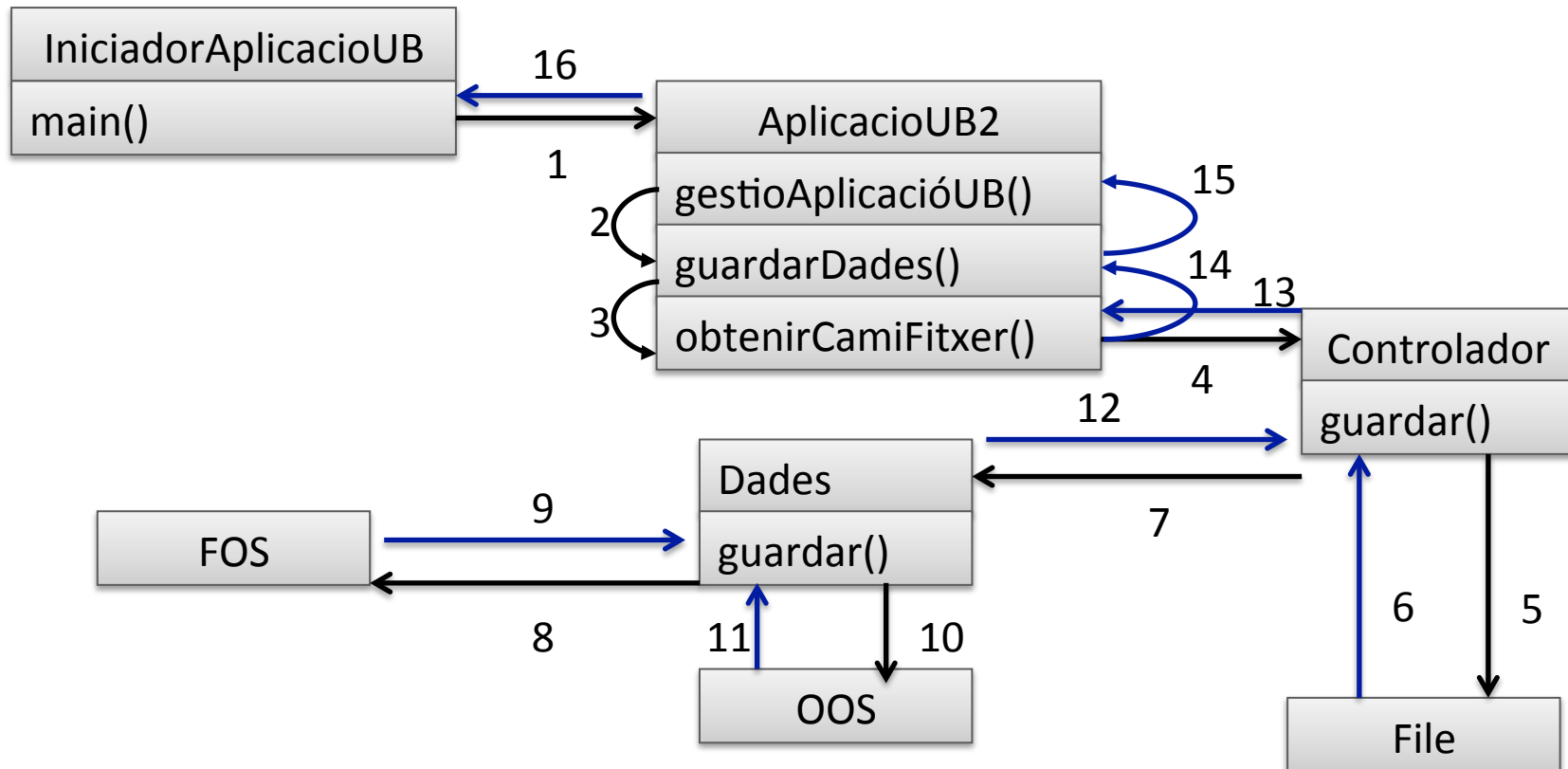
- Dubtes correccions L1
- Preguntes memòria L2
- **Diagrama de flux**

# Preguntes memòria L2: diagrama de flux (concepte i definició)

“A flowchart is a type of diagram that represents an algorithm, workflow or process, showing the steps as boxes of various kinds, and their order by connecting them with arrows”

(<https://en.wikipedia.org/wiki/Flowchart>; 2018)

# Preguntes memòria L2: diagrama de flux (exemple: guardar)



\*No considerem les excepcions, perquè es surten del flux "normal" del programa

## Preguntes memòria L2: diagrama de flux per afegir fitxer a la biblio.

- Per la pràctica L2 us demanem que feu el diagrama de flux per afegir un fitxer multimèdia a la biblioteca de fitxers.

# Preguntes memòria L2:

## sobrecàrrega

- Expliqueu si es podria fer servir la sobrecàrrega de mètodes a la classe Controlador i, en cas afirmatiu, detalleu com
- Recordatori:
  - **Sobrecàrrega:** mètode amb el mateix nom, però amb signatures diferents; per ex. un constructor