

# PROGRAMACIÓ ORIENTADA A OBJECTES

## TRIMESTRE 2 – CURS 2021/2022

### PRÀCTICA 4

- Objectiu:** SWING. Creació d'interfícies gràfiques d'usuari més complexes. **Posició manual sense ús del WindowBuilder**
- Durada:** Dues sessions
- Lliurament:** Llistat imprès dels fonts i penjar el projecte al Moodle
- Important:** Imprescindible per aprovar la pràctica seguir les pautes de l'enunciat
- Data Lliurament:** El 23 de Març a les 8:00h (dia examen)

#### JOC DE LA VIDA

Cal dotar d'interactivitat al joc que vàreu programar a la primera pràctica de l'assignatura.

La lògica del problema la teniu explicada a l'enunciat de la pràctica 1.

#### Especificacions del joc

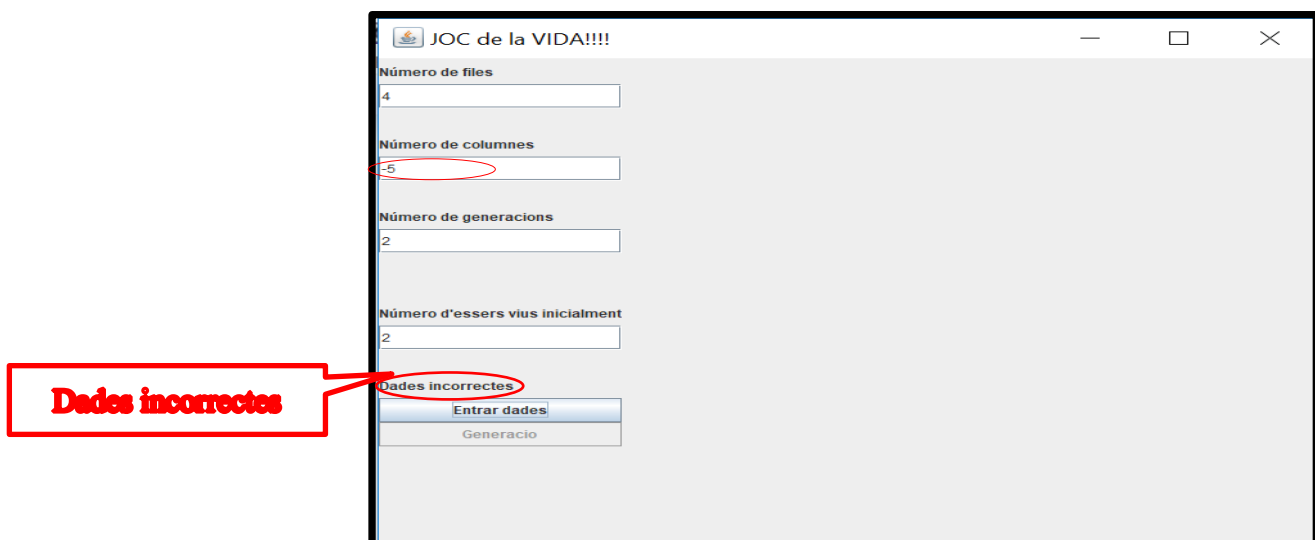
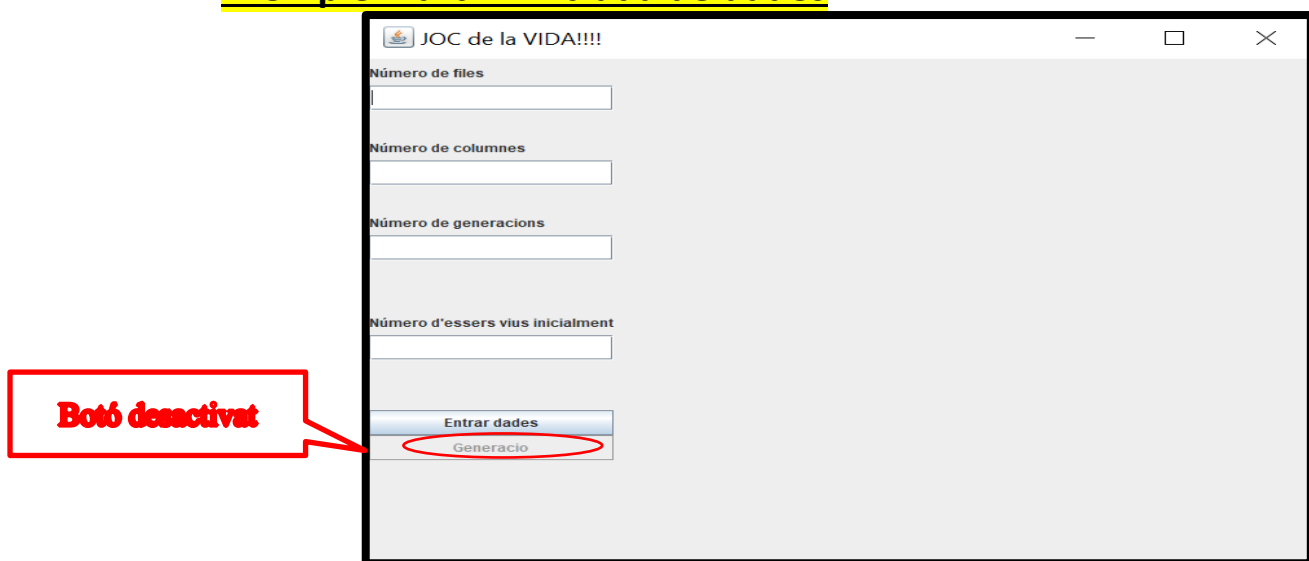
- Les dimensions del taulell són demanades a l'usuari, **han de tenir un valor sempre positiu** major a zero i **no necessàriament han de ser iguals, és a dir el taulell pot ser rectangular.**
- La quantitat d'organismes vius inicialment també es demana a l'usuari. Aquest valor sempre ha de ser positiu i menor al nombre de caselles del taulell. Les caselles que contindran aquests organismes vius seran generades **aleatòriament**. Controleu les repeticions de caselles. Per generar un número aleatori useu la funció Math.random() o la classe Random.
- L'usuari abans d'iniciar-se el joc ha d'indicar el número de **generacions** que vol assolir, sempre ha de ser un valor positiu. L'aplicació a l'assolir-les **ha de parar**. Si abans d'assolir el número màxim de generacions el número d'organismes vius que hi ha al taulell és zero o bé, dues generacions consecutives no generen cap canvi, l'aplicació **també ha de parar**.
- **Quan finalitza el joc** es para tota la interactivitat i **NO cal mostrar cap resultat estadístic.**

#### Especificacions de la GUI

- L'aplicació té només un escenari. En una part d'aquest, a l'esquerra de la finestra, apareixen els elements adients per sol·licitar a l'usuari les dades per parametritzar el joc. Quan l'usuari les ha especificat i són correctes, es crea el taulell. Aquest està ubicat a la part dreta de la finestra. **S'ha d'activar un element per anar creant generacions**, una a una segons vagi interactuant l'usuari, aquest element ha de ser un objecte de la classe JButton. Cada cop que l'usuari el premi es crearà una nova generació.

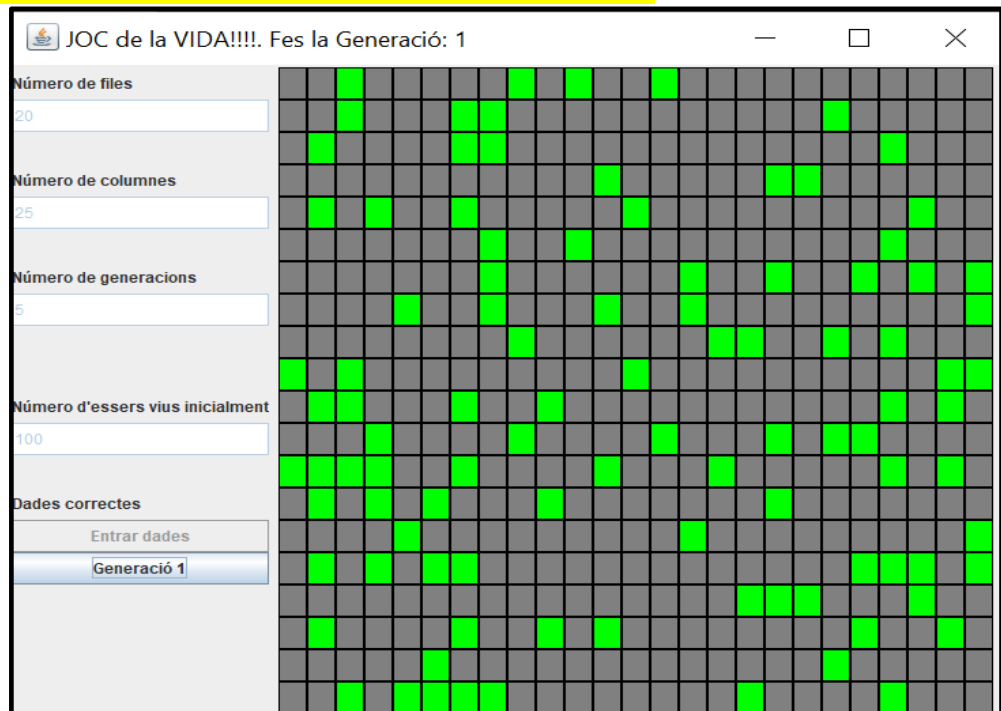
- Cal controlar la correctesa de les dades entrades, mentre no siguin correctes s'ha d'insistir. Els objectes gràfics adients per entrar aquesta informació són objectes de la classe JTextField. Aquests sempre acompanyats d'objectes de la classe JLabel per indicar quina informació s'està sol·licitant.
- Un objecte de la classe JButton activarà la recollida i comprovació de la correctesa de les dades. Si són correctes, s'activarà el botó per a què l'usuari vagi creant generacions, una en cada premuda. Si no ho són s'ha d'indicar en una etiqueta que hi ha valors incorrectes.
- Aprofiteu el títol del Frame per a mostrar quina generació és l'actual.
- La interfície gràfica s'ha de **dissenyar usant gestors de posicionament** per aconseguir una adaptació dels components quan canvia la mida de la finestra.

### Exemple. Part 1. Entrada de dades



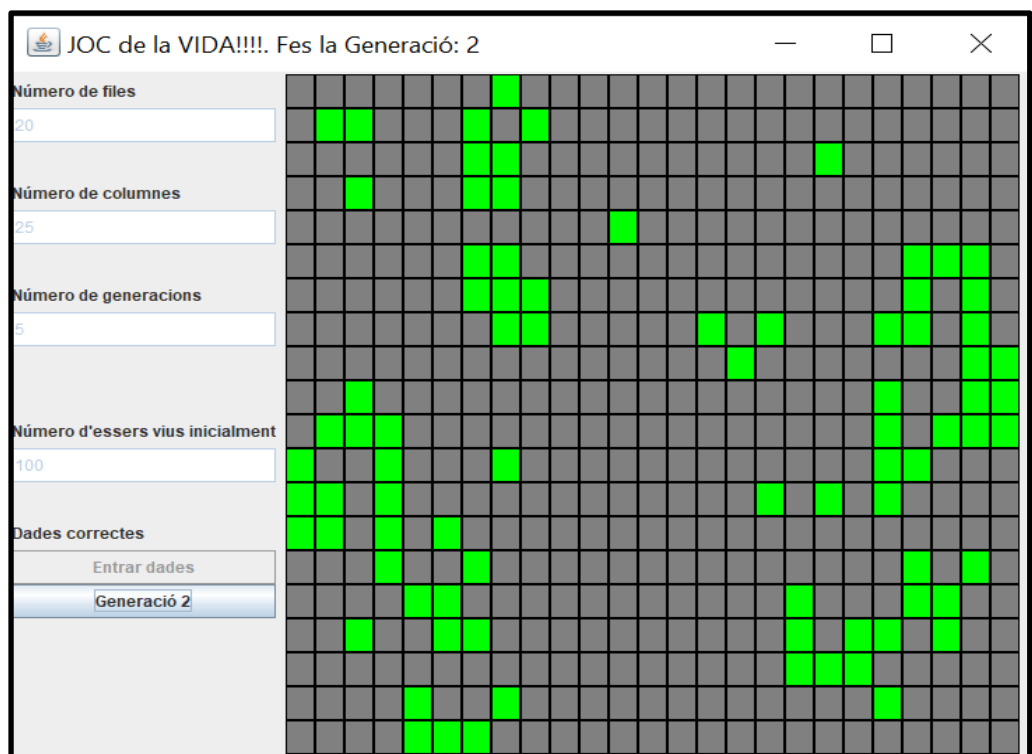
## Exemple. Part 2. Amb dades entrades correctes

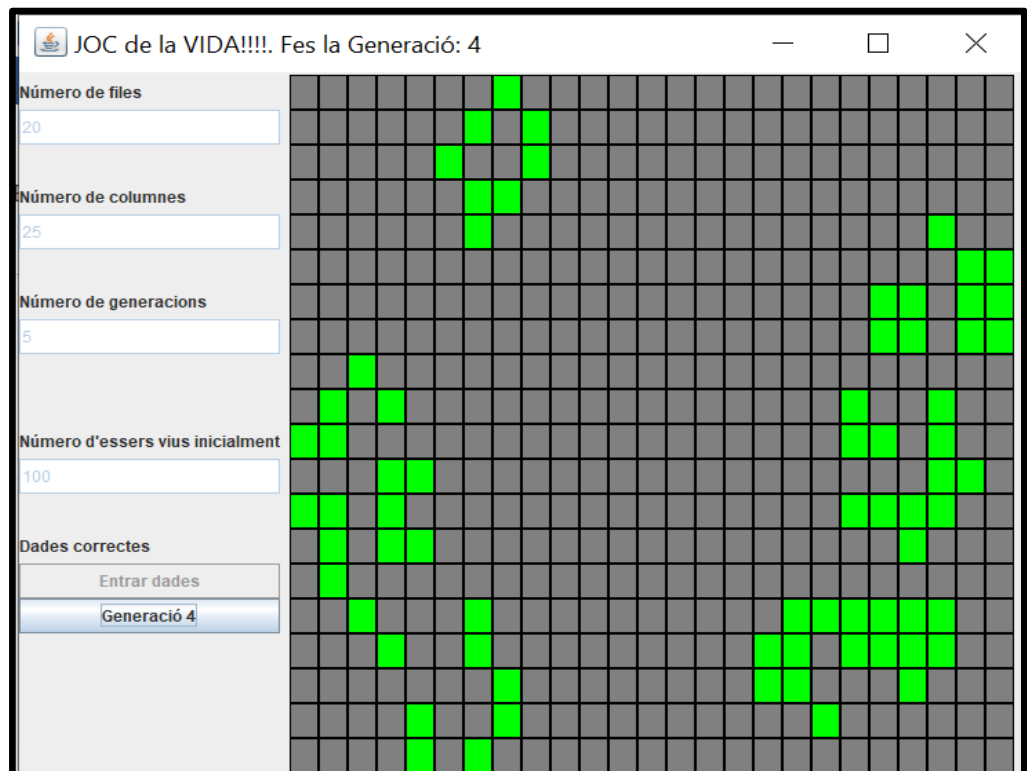
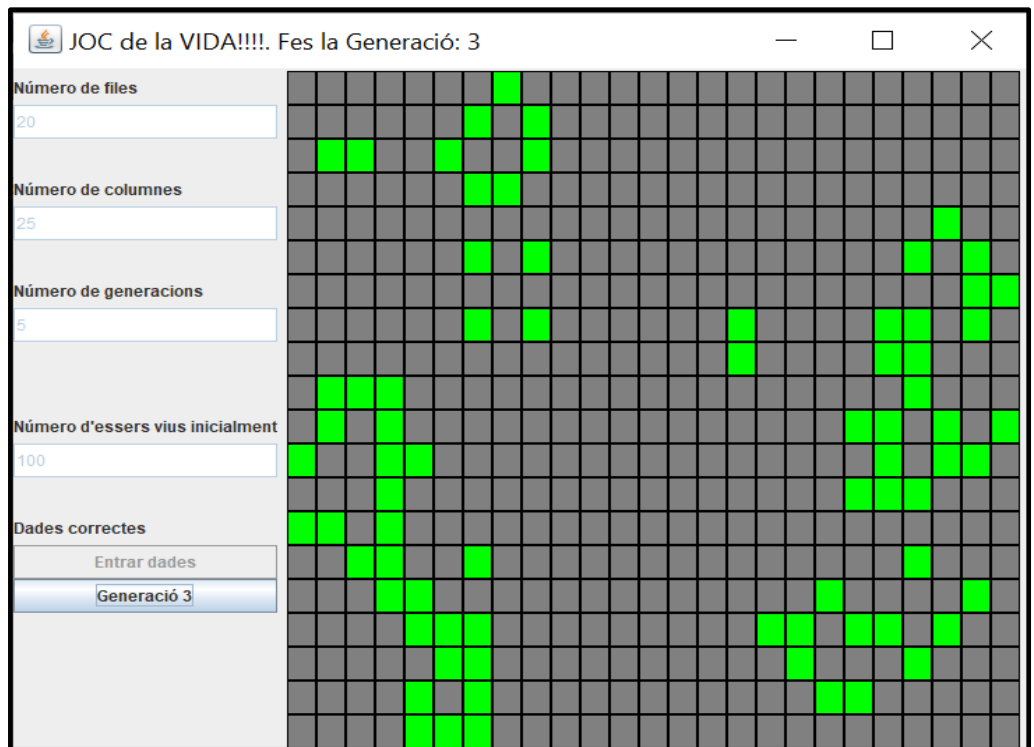
En el moment en què les dades són correctes es crea el taulell i els components de la part esquerra queden deshabilitats. S'activa el botó de fer generació

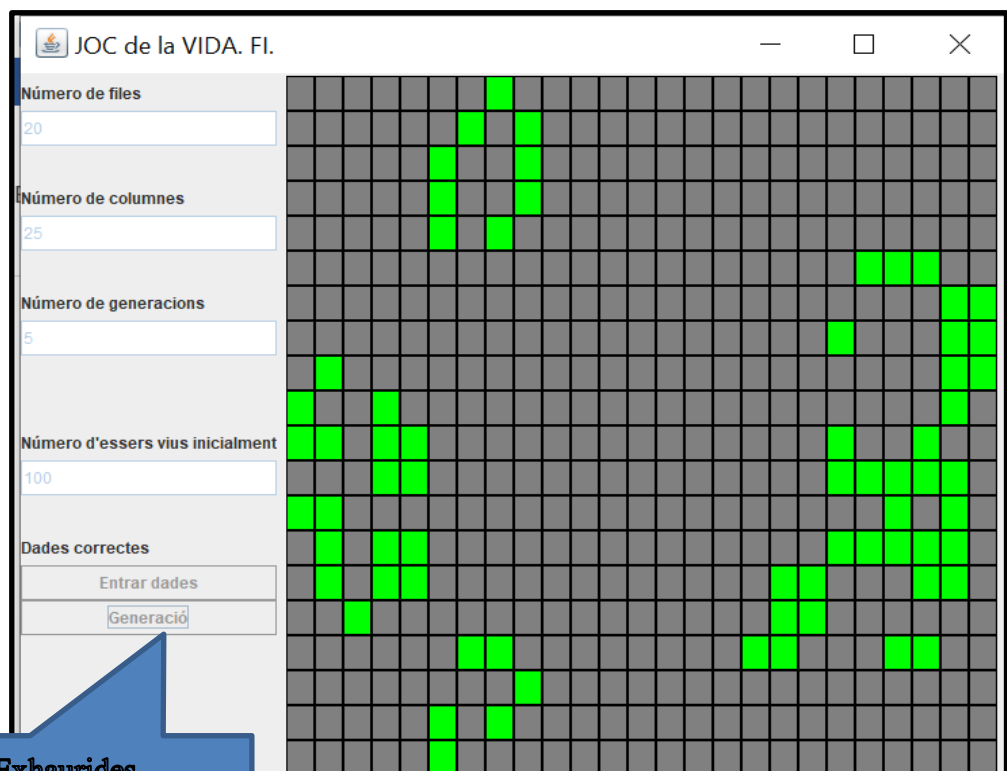
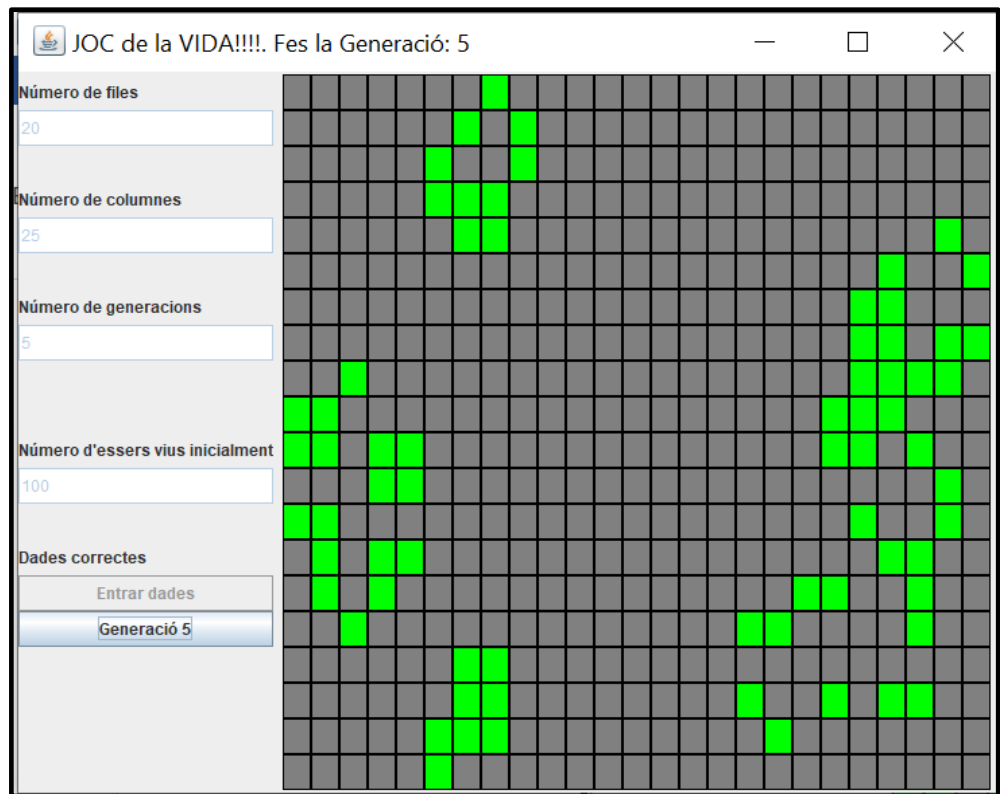


Les caselles amb un organisme viu tenen fons verd i les que no fons gris.  
El títol de la finestra s'actualitza en cada generació.

L'etiqueta del botó s'actualitza en cada premuda, indicant el número de la propera generació a fer



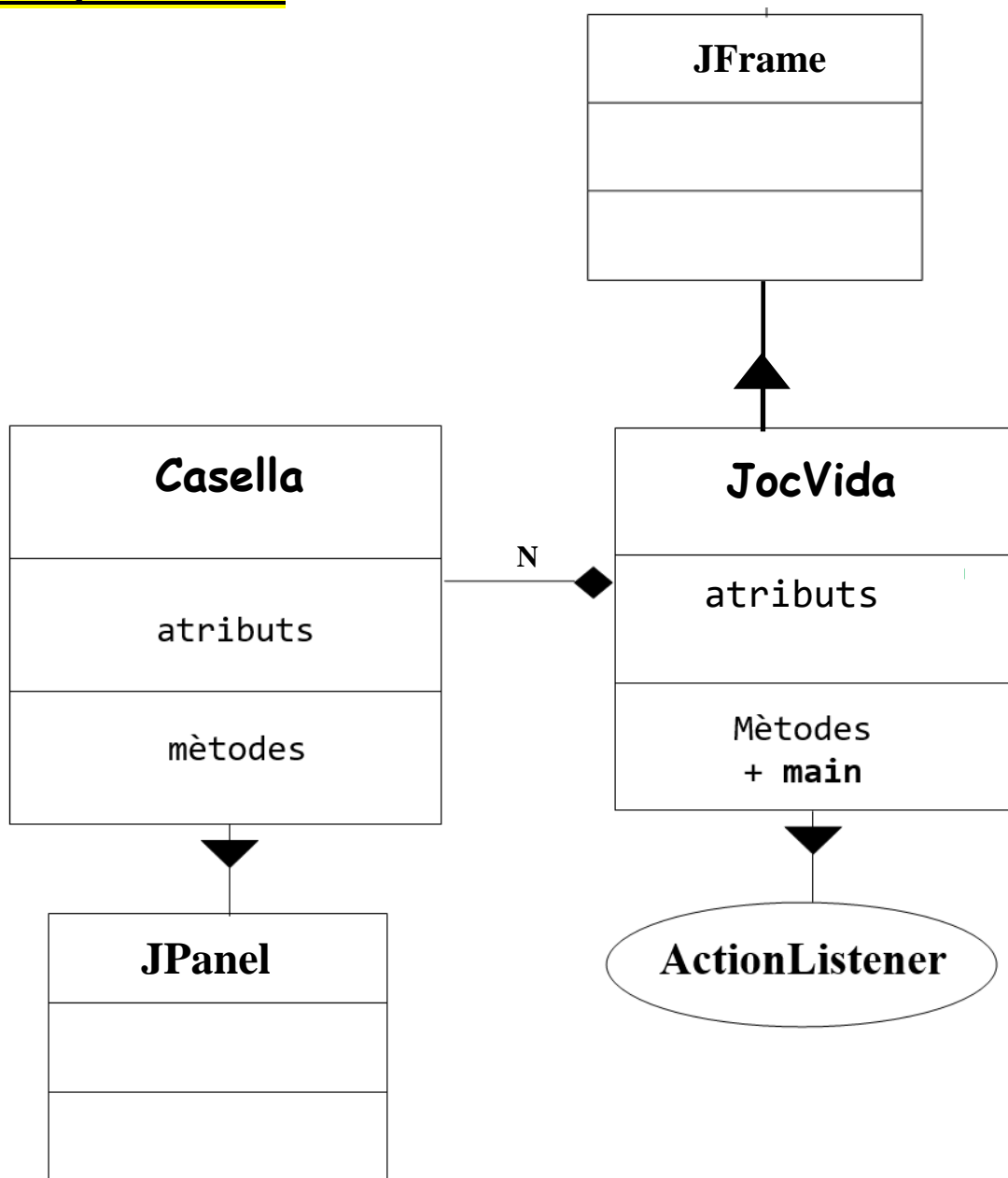




Joc Acabat. Exhaurides les generacions. Recordeu que s'han de contemplar diferents acabaments

Quan s'assoleix la darrera generació el joc finalitza. S'ha de tancar l'aplicació prement la X de la cantonada amunt dreta.

## Disseny de classes



La relació d'herència entre la interfície ActionListener i la classe JocVida indica que l'escoltador dels events és el propi contenidor d'alt nivell.

## Implementació de les classes

### ➤ Primer cal escriure la classe Casella

Un objecte d'aquesta classe representa una casella del taulell del joc de la vida. Serà un component JPanel que s'especialitzarà per controlar el color de fons en funció de si representa una casella que conté o no vida. Aquests objectes no han de gestionar cap tipus d'event i en conseqüència no s'ha optat perquè siguin botons. Tot i que els objectes de la classe JPanel són contenidors no s'usaran com a tals, no ubicaran cap altre component.

```
public class Casella extends JPanel{
    .....
}
```

Els objectes d'aquesta nova classe no tindran cap atribut propi. Cal afegir-li només **mètodes**, concretament:

- **Constructor** sense paràmetres que crea una casella sense vida, buida.
- Mètodes per consultar, modificar i alternar l'estat d'una casella referent a si està buida o té un organisme viu.

### ➤ **Segon cal escriure la classe JocVida**

La classe ha de ser una extensió de la JFrame i implementar la interfície **ActionListener** per a que pugui esdevenir un escoltador de la interactivitat feta per l'usuari.

```
public class JocVida extends JFrame implements ActionListener{
    .....
}
```

Els objectes d'aquesta classe tindran **atributs privats**:

- **Obligatòriament un atribut per a cadascun dels components gràfics de la GUI.**
- **Una matriu d'objectes de tipus Casella.** La dimensió, files i columnes seran dades entrades per l'usuari.

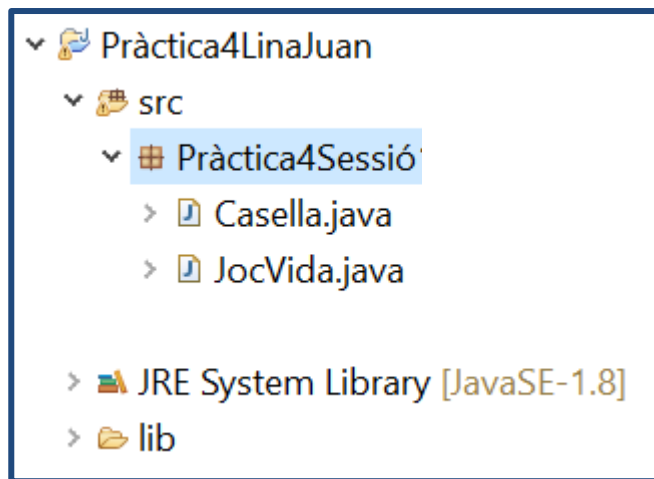
A més hi podeu afegir tants nous atributs com considereu necessaris.

La classe tindrà els següents **mètodes**, tot i que podreu afegir-ne els que considereu necessaris per la codificació del joc:

- El mètode **main()**.
- **Constructor** per crear la interfície, ajudeu-vos d'un mètode privat **initComponents**.
- Implementar el mètode **actionPerformed** a la que obliga la implementació de la interfície per donar resposta als events dels botons.
- Els **mètodes necessaris per dotar de la funcionalitat al joc, són els que ja teniu**. Heu d'usar els de la vostra solució amb POO, Pràctica 1 Sessió 2. Incorporar aquells mètodes que ja teniu fets, en alguns casos fent petites adaptacions.

## **Organització**

Seguireu la mateixa estructura que la usada a les pràctiques precedents. Cal crear un nou projecte amb nom **Pràctica4Cognom1Nom1&&Cognom2Nom2**. El projecte tindrà un únic paquet on hi ubicareu les dues classes demanades.



### Què se us subministra?

L'enunciat amb les especificacions i línies a seguir per el desenvolupament del disseny de classes.

### Què s'ha de lliurar i com?

S'ha de lliurar la carpeta que conté el projecte Eclipse amb el vostre desenvolupament de la pràctica. La carpeta s'ha de lliurar amb tot el seu contingut i comprimida amb ZIP o RAR.

També s'ha de lliurar un **llistat en paper** del codi desenvolupat (no la classe Prova). El format de lliurament d'aquest codi ha de seguir el patró indicat en la presentació de l'assignatura: amb portada, índex, número de pàgina, tabulació ... En **aquest llistat** cal que indiqueu:

- la distribució de la feina entre els dos estudiants. És a dir, el grau de participació de cada membre del grup en la realització d'aquesta activitat.
- Si el programa no funciona cal que indiqueu quina/es parts no funcionen explicant que és el que passa.

### On s'ha de lliurar?

El lliurament del projecte es farà a través de la plataforma Moodle i no s'acceptarà cap altra via. Feu atenció a la data i hora límit.

### Quan s'ha de lliurar?

El lliurament es podrà fer fins el **dia indicat a sota**. Tingueu present que a partir d'aquesta hora el sistema bloquejarà, de manera automàtica, la possibilitat de lliurament.

**Lliurament Moodle Grups 101 i 102 → 23 de Març a les 8h**

**Lliurament Paper Grups 101 i 102 → 23 de Març a les 11h - Inici examen no més tard**



### **Pauta de correcció:**

En la correcció de l'exercici d'aquesta sessió es valorarà:

- 1.- La programació "lògica" del joc, seguint les pautes indicades en l'enunciat, amb els mètodes i atributs demanats. **Ja ho teniu fet** de la pràctica 1.
- 2.- La programació de la GUI seguint les pautes també indicades.
- 3.- Com sempre en la seva escriptura es valorarà també: l'aplicació dels **esquemes** adients (recorregut i cerca) i l'**eficiència** en quan a no fer càlculs repetits amb unes mateixes dades.

La qualificació dependrà dels ítems indicats prèviament. **Obligatòriament heu de seguir les pautes de programació** donades en l'escriptura del programa. La vostra solució s'ha d'ajustar a l'enunciat de l'exercici.

Si el programa no funciona la qualificació màxima que es pot assolir és un 4.

**NO es considerarà correcte** cap lliurament que s'hagi obtingut usant el WindowBuilder.

**En aquest cas la qualificació serà de 0.**