

PROGRAMACIÓ ORIENTADA A OBJECTES

TRIMESTRE 2 – CURS 2021/2022

PRÀCTICA 1. SESSIÓ 2

Objectiu: Treball amb taules de dues dimensions. Les matrius
POO. Conceptes bàsics

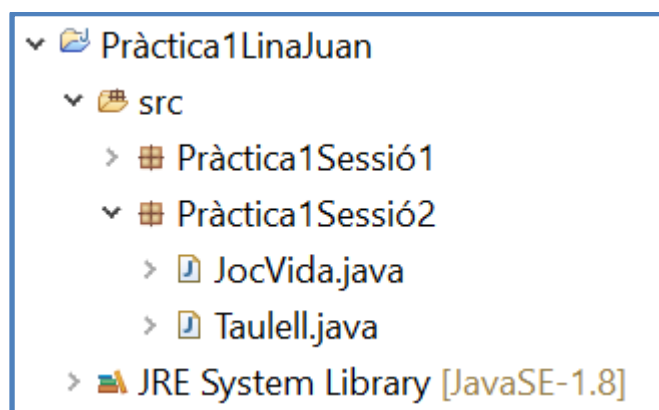
Durada: Una sessió

Lliurament: Llistat imprès dels fonts i penjar el projecte al Moodle

Data Lliurament: Abans de la pràctica 2. **3 de Febrer a les 23:50h**

Joc de la Vida. Organització

Aquest nou plantejament l'ubicareu dins d'un altre paquet corresponent a la sessió 2, dins del mateix projecte corresponent a la pràctica 1 de l'assignatura.



El programa tindrà dues classes, les dues les ubicareu dins del mateix paquet:

- 1.- la classe anomenada **Taulell** que tindrà tots els mètodes i atributs necessaris per la gestió del taulell del joc, i
- 2.- la classe **JocVida**, que tindrà el mètode `main` i els procediments d'ús privat per aquelles parts que per sí soles tenen una funcionalitat clara o bé d'ús repetitiu.

Primer cal escriure la classe Taulell

Aquesta classe tindrà els següents **atributs**, tots ells de visibilitat privada:

- Un **atribut1** per representar el taulell de joc, evidentment serà una matriu de booleans
- Un **atribut2** per emmagatzemar el número de files que té el taulell de joc
- Un **atribut3** per emmagatzemar-hi el número de columnes del taulell de joc
- Un **atribut4** per comptabilitzar el número d'essers vius que té el taulell en tot moment (cada cop que es modifica l'estat d'una casella aquest atribut s'ha d'actualitzar).

Doneu a cada atribut el nom que considereu més adient.

I els següents **mètodes** de visibilitat pública (si no s'indica el contrari):

- Un **mètode constructor** per crear el taulell de joc a partir de dos paràmetres que indicaran les dimensions d'aquest. Tots els atributs de l'objecte construït han de quedar inicialitzats. Supposeu que els valors dels paràmetres **sempre** tenen un valor correcte. Aquest mètode ha de crear el taulell de joc (atribut matriu) **sense** organismes vius, és a dir sense vida, per això invoqueu al mètode següent per la inicialització.

```
public Taulell(int n, int m)
```

- Un **mètode que posi el taulell sense cap organisme viu**. Aquest mètode **no ha de crear** el taulell, únicament ha de treure tots els organismes i posar l'atribut4 a 0 (no usar l'operador **new**).

```
public void treureVida()
```

- Un mètode que **ubica N organismes al taulell**, essent N un paràmetre del mètode. Useu la funció **Math.random()** o la classe **Random** per generar **quines** caselles han de tenir vida. Recordeu que s'han d'ubicar en exactament N caselles i no poden estar repetides. El valor del paràmetre N **sempre** és correcte.

```
public void posarVida(int N)
```

- Un mètode per **mostrar a pantalla el taulell** amb tot el seu contingut en format matriu.

```
public void mostraTaulell()
```

- Un mètode **ferGeneracio** sense paràmetres que ha de crear una nova generació a partir de l'actual. Aquest mètode tindrà una variable local de tipus matriu, del mateix tipus que l'atribut1 de l'objecte, que recollirà els resultats així com vagi iterant. Un cop finalitzada la iteració el contingut d'aquesta variable local es copiarà sobre l'atribut1. Per fer la implementació d'aquest mètode invoqueu als mètodes següents que han de tenir una visibilitat privada: **quantasVeines** i **copiar** descrits a continuació.

```
public boolean ferGeneracio()
```

El retorn serà true si la nova generació és diferent de l'anterior, fals en cas contrari. Per això cal comparar les dues matrius (aux i joc) abans de fer el traspàs de dades.

- Un mètode **privat quantasVeines** que donada una casella del taulell, calcula i retorna el nombre de caselles adjacents que tenen organismes vius. Una casella ve donada per la seva ubicació dins del taulell, fila i columna. Considereu sempre correctes el valor dels paràmetres del mètode.

```
private int quantasVeines(int fil, int col)
```

- Altre mètode **privat** que fa una **còpia** de les referències de les matrius, la donada en el paràmetre sobre l'atribut1 de l'objecte. El mètode té només un paràmetre.

```
private void copiar(boolean[][] origen)
```

- Un mètode **getAtributX** pels atributs 2, 3, 4 per consultar el valor d'aquests atributs privats d'un objecte Taulell.

Nota: cap d'aquests mètodes pot tenir operacions de lectura, les dades necessàries li arriben via paràmetres. Només el mètode que ha de visualitzar el taulell pot fer ús de les operacions d'escriptura (System.out.print ó System.out.println) i també, el mètode que ubica aleatòriament éssers vius pot mostrar a pantalla la casella on els ubica.

Segon cal escriure la classe JocVida

Com ha mínim ha d'incloure el punt d'entrada en execució del programa, el procediment **main**.

En aquesta classe, s'ha de:

- sol·licitar a l'usuari mitjançant operacions de lectura totes les dades que requereix el joc, sempre insistint mentre les dades siguin incorrectes,
- emmagatzemar el número d'organismes vius després de cada evolució, informació que s'ha de mostrar al finalitzar el joc, a més d'indicar també, quina ha estat l'evolució que ha tingut més vida,
- i gestionar la continuïtat en jugar o no novament amb el mateix taulell i número d'organismes vius que hi ha inicialment. Aquests organismes seran novament ubicats aleatòriament al taulell. **No s'ha de crear novament el Taulell**, s'ha de buidar i ubicar novament els organismes vius.

Es valorarà positivament l'escriptura de procediments privats per aquelles parts que per sí soles tenen una clara funcionalitat i també per aquelles que s'han d'executar varies vegades amb l'objectiu de reaprofitar codi.

Serà en aquesta classe **on es controlarà la correctesa de totes les dades entrades per l'usuari** i no pas als mètodes de la classe Taulell, que les rebran via paràmetre i les consideraran correctes.

Denoteu que tota la part que fa referència a modificacions i consultes del taulell les haureu de fer invocant als mètodes de la classe Taulell, tots ells amb visibilitat pública.

Aquesta classe no tindrà atributs, únicament el mètode **main** i procediments privats per ajudar aquest main. **Ni el main ni els procediments poden fer ús de variables de tipus matriu i únicament el main declararà i crearà un objecte de la classe Taulell.**

```
public static void main(String args[]) {  
    Taulell t;  
    // instruccions  
}
```

Què se us subministra?

L'enunciat amb les especificacions i línies a seguir per el desenvolupament del joc.

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 → 3 de Febrer a les 23:50h

Pauta de correcció:

En aquesta pràctica de dues sessions resoldreu un **únic exercici per sessió**.

Cadascun té un pes de 5 punts però la correcció es farà sobre 10 i després es ponderarà.

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

- 1.- La correctesa en l'escriptura de la classe Taulell, seguint les pautes indicades a l'enunciat. **L'escriptura d'aquesta classe té un pes de 50%**. Com sempre en la seva escriptura es valorarà també: l'aplicació dels **esquemes** adients (recoregut i cerca), pautes de programació en l'accés a les caselles adjacents, l'**eficiència** en quan a no fer càlculs repetits amb unes mateixes dades, que no hi hagi cap operació de lectura en cap dels seus mètodes i que només visualitzin dades a pantalla els dos mètodes indicats en l'enunciat.
- **La classe Joc té també un pes dels 50%**. És valorarà també: la **robustesa**, és a dir, el control de què les dades demanades a l'usuari siguin correctes, l'anàlisi descendent (escriptura de procediments d'ús privat del main), ús de les estructures de dades indicades en l'enunciat per emmagatzemar dades estadístiques: els vectors.

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 3.