

Programació Avançada

Pràctica 1. Curs 2024/2025

Centre universitari adscrit a la



Rosa Herrero

Consultes presencials els divendres de 10.00 a 12.00 hores al despatx 14.

Consultes virtuals a través del correu electrònic rHerrero@tecnocampus.cat

Si us plau, per consultes virtuals envieu no envieu tot el projecte, només aquella part que tingueu dubte i en format PDF.

Normativa

- Cal treballar en parelles, i cal que tots dos alumnes siguin del mateix grup de classe (G101, G102, G103, G104 o G105).
- No s'admeten lliuraments per correu electrònic o demores sense cap justificació (citació judicial o assumpte mèdic). Una pràctica no entregada sense justificació compta com un 0.
- Les pràctiques s'han de lliurar a través de l'aula virtual.

Sistema d'avaluació

Pràctica 1 6%

Pràctica 2 6%

Pràctica 3 6%

Pràctica 4 6%

Pràctica 5 6%

Per aprovar l'assignatura és necessari que l'estudiant:

- com a mínim, hagi lliurat una de les dues primeres pràctiques, i què la qualificació sigui superior o igual a 4.
- com a mínim, hagi lliurat dues de les tres darreres pràctiques, i què la qualificació de cadascuna sigui superior o igual a 4.

Si no es donen aquestes premisses l'assignatura quedarà suspesa i en aquest cas sense possibilitat de recuperar-la.

Pràctica 1: Simulació d'una partida d' escacs

Objectius:

- Treballar amb colleccions genèriques <E>.
- Guardar en fitxers

Llenguatge: JAVA

Duració: 2 sessions = 2 setmanes

Projecte inicial: només teniu la interfície a copiar.

Cal llegir la normativa de practiques.

Lliurament: 6/10/2024

Pràctica 1: Simulació d'una partida d'escacs

Descripció:

L'objectiu d'aquest exercici és simular una partida d'escacs, on cada jugador tindrà una llista de torns jugats, i cada torn contindrà una llista de les peces que continuen vives al tauler en aquell moment. Les peces seran objectes d'una classe que indicarà el tipus de peça (Rei, Dama, Torre, Cavall, Alfil o Peó) i la seva posició actual al tauler (matriu de 8x8).

A cada torn, el jugador mourà una peça i es comprovarà si s'ha capturat la peça de l'oponent per modificar la seva llista de peces vives, a continuació serà el torn de l'altre jugador.

Important: no cal comprovar que els moviments siguin vàlids, només realitzar els moviments.

Pràctica 1: Simulació d'una partida d' escacs

Al principi de l'execució es mostrarà un menú principal on es preguntarà:

- si es vol jugar una partida nova,
- reproduir una partida a partir d'un fitxer
- o sortir de l'execució.

Després de cada partida es retornarà a aquest menú.

En cap moment cal
comprovar que la
jugada sigui correcte

Pràctica 1: Partida nova

Si es vol jugar una partida nova:

- S'iniciaran dos jugadors (blanc i negre) amb totes les peces inicials a la seva posició corresponent
la classe Jugador tindrà un llistat de peces vives
- El jugador blanc indicarà quina peça vol moure i la nova posició,
la nova posició cal que no estigui ocupada per cap peça blanca
- Si la nova posició té una peça negra, aquesta serà eliminada.
- A continuació serà el torn del jugador negre.

Durant el joc es guardaren tots els torns dins d'un llistat genèric (classe Torns), i en finalitzar la partida es guardarà en un fitxer.

Pràctica 1: Reproduir una partida a partir d'un fitxer

A partir d'un fitxer es reproduirà una partida:

- Demanarem el nom del fitxer i llegirem tots els torns.
- S'iniciaran tots dos jugadors amb totes les peces,
- A partir del llistat de torns es realitzaran, i visualitzarà, el moviment de cada peça fins acabar el joc.

Sempre comença el
jugador blanc

Pràctica 1: convertir el torn

Ja sigui que el jugador ha informat d'un torn per pantalla o que s'hagi llegit del fitxer, caldrà entendre el missatge en una instància de la classe Torn.

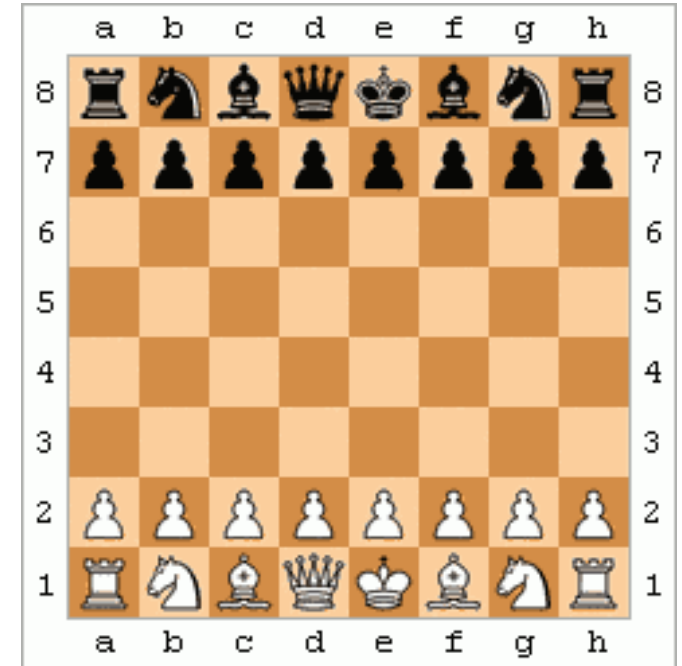
- Les columnes del tauler d'escacs es representen amb les lletres de la A a la H (majúscules).

Aquestes lletres s'han de convertir en un número enter.

Podem utilitzar el valor ASCII de les lletres per fer aquesta conversió on 'A' correspon a la columna 0.

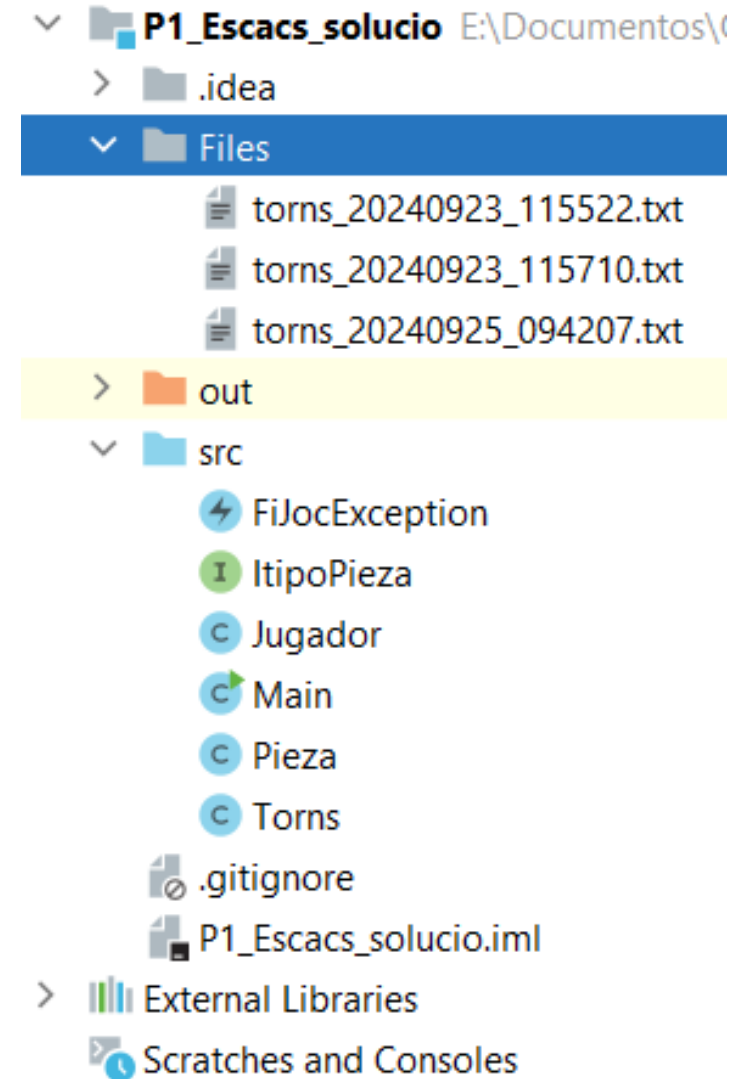
El codi ASCII per 'A' és 65, així que podem restar 64 per obtenir la columna desitjada.

- La fila es representa com un número del 1 al 8.



Projecte

- Crea un projecte amb el nom **P1_Alumne1_Alumne2**



Classe Torns

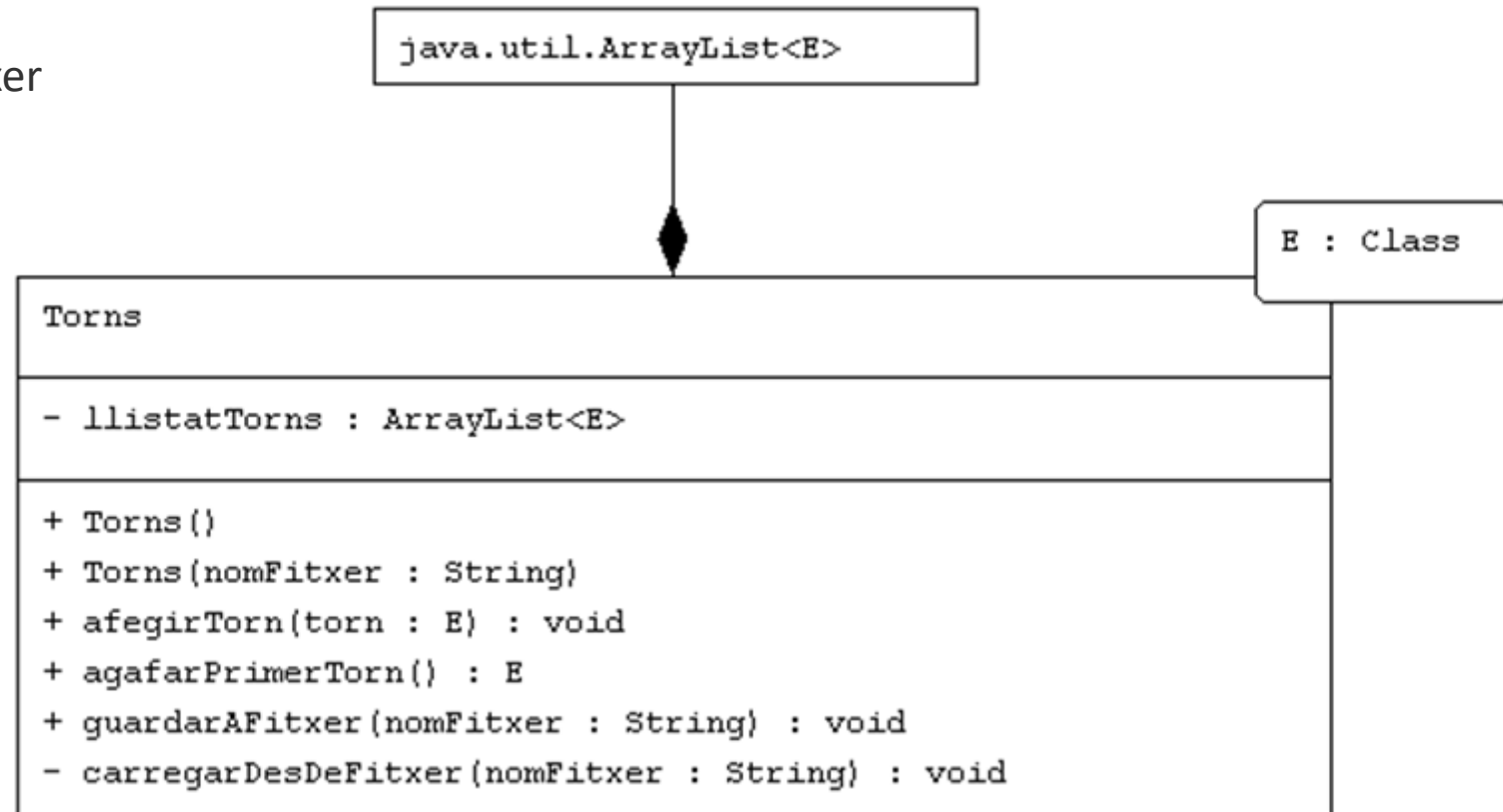
Utilitzarem una classe Torns per

- Guardar tots els torns d'una partida
- Reproduir una partida a partir d'un fitxer

IMPORTANT: Cal utilitzar llistat genèric:

```
public class Torns <E>{  
    private ArrayList<E> llistatTorns;
```

IMPORTANT: Cal seguir el diagrama



Classe Torns

Al finalitzar una partida nova es guardarà tots els torns de la partida jugada:

- **Torns:** constructor per defecte que inicialitza el llistat de torns.
- **afegirTorn:** afegeix un torn a l'última posició del llistat.
- **guardaraFitxer:** guarda tots els torns de la partida en un fitxer del tipus txt, podria retornar una excepció del tipus IOException.

Per poder reproduir una partida a partir d'un fitxer:

- **Torns(nomFitxer):** invoca el mètode privat llençarà una excepció si no s'ha pogut llegir o si la llista és buida.
- **carregarDesDeFitxer:** mètode privat que llegeix un fitxer txt i inicialitzar el llistat de torns a partir del fitxer.
- **agafarPrimerTorn:** elimina i retorna el primer element del llistat (que serà el torn a realitzar), si no hi ha més retornarà una excepció del tipus NoSuchElementException.

Interfície ItipoPieza

- Afegeixiu al teu projecte la següent interfície:

```
public interface ItipoPieza {  
    // mètodes abstractes: aquí detallem les capçaleres,  
    // les classes que implementen la interfície, hauran d'implementar aquests mètodes  
    public abstract char getTipus();  
    public abstract int getFila();  
    public abstract int getColumna();  
    public abstract void setPosicion(int fila, int columna) throws RuntimeException;  
    // si la posició no és correcta cal llançar una excepció  
    public abstract boolean fiJoc() ;  
}
```

IMPORTANT: no modifiqueu aquesta interfície

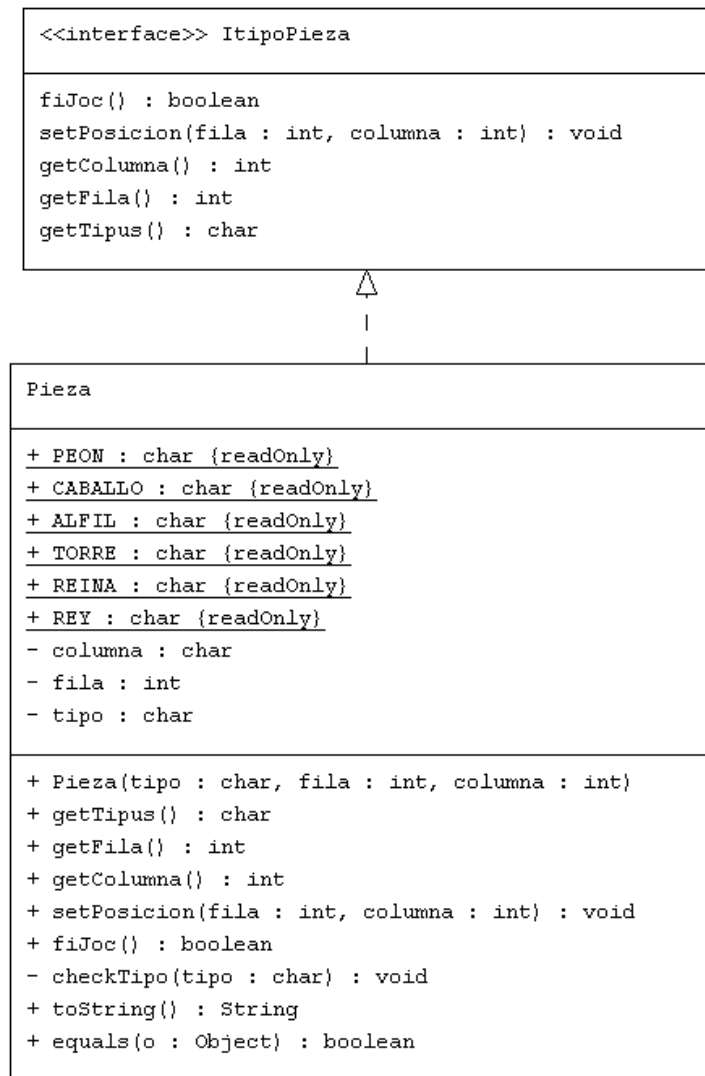
Classe Pieza

- Cal implementar una classe anomenada Pieza que implementi l'anterior interfície:

- Necessitarem 3 atributs objectuals (cada objecte tindrà un valor propi): el tipus de figura, la fila i la columna.

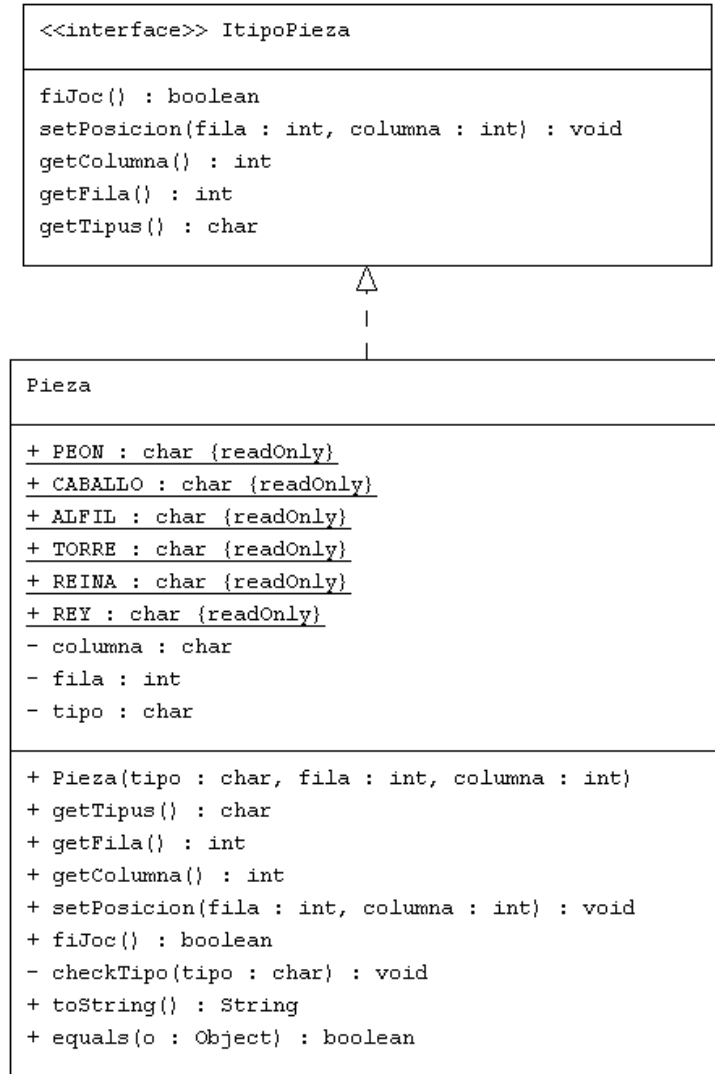
En el cas del escacs, les columnes es representen amb una lletra (veure imatge p10).

- Cal definir constants amb tots els tipus de figures.



Classe Pieza

- setPosicio retornarà una excepció si els valors introduïts no són correctes.
- fiJoc retornarà true si l'atribut tipo és REY, s'utilitzarà per saber si el rei s'ha mort.
- checkTipo és un mètode privat que s'invoca des del constructor per comprovar el valor donat, si no és correcte cal retornar una excepció.



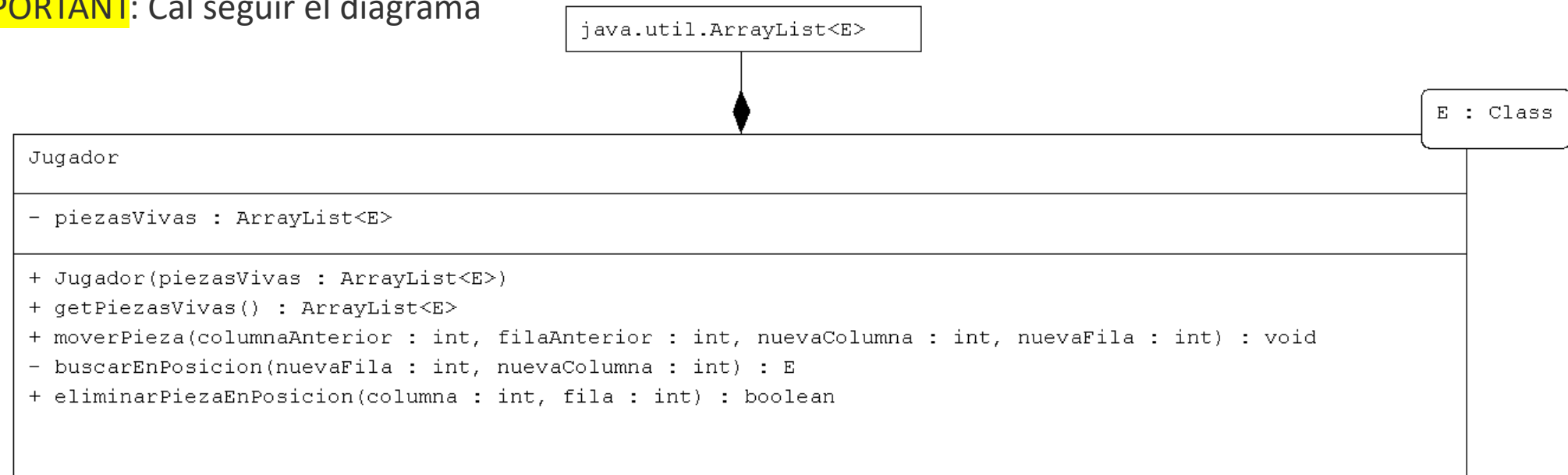
Classe Jugador

- La classe Jugador contindrà un llistat de peces vives al tauler, però no farà referència a la classe Pieza, és a dir, cal que el llistat sigui genèric per qualsevol classe que implementi la interfície ItipoPieza.

IMPORTANT: Cal utilitzar llistat genèric:

```
class Jugador<E extends ItipoPieza> {  
    private ArrayList<E> piezasVivas;  
}
```

IMPORTANT: Cal seguir el diagrama



Classe Jugador

- **Jugador(pecesInicials):** Implementa un constructor que inicialitzi l'objecte a partir d'un llistat de peces predefinides.
- **buscarEnPosicion:** Implementa un mètode privat que retorni la peça d'una posició donada aquest mètode s'invocarà al eliminar o al moure.
- **moverPieza:** Implementa un mètode per moure una peça d'una posició a una altra, sinó es troba llança un RuntimeException. Cal invocar al mètode privat.
- **eliminarPiezaEnPosicion:** Implementa un mètode per eliminar una peça, retorna un booleà segons si s'ha eliminat o no, en cas que s'elimini el rei cal llançar una excepció del tipus FijocException. Cal invocar al mètode privat buscarEnPosicion i el mètode fijoc de la interfície.
- **getPiezasVivas:** Implementa un getter que retorni el llistat, s'utilitzarà al Main per mostrar les peces al tauler.

Excepció per finalitzar el joc

- Crea una classe personalitzada FiJocException que hereti de la classe Exception, per informar que s'ha eliminat el rei. El constructor per defecte invoca al super amb un missatge personalitzat.
- A la classe Jugador quan s'elimini el rei (o sabrem gràcies al mètode fiJoc de la interfície ITipoPieza) es llançarà una excepció del tipus FiJocException per avisar que s'ha acabat el joc.
- Cal que al Main es controli la excepció de forma adequada.

Diagrama i mètodes orientatius:

<<utility>> Main

- scanner : Scanner {readOnly}

+ main(args : String[]) : void

- mostrarMenu() : void

- jugarNovaPartida() : void

- reproduirPartida() : void

- llegirTorns() : Torns<String>

- tornToPosition(torn : String, p1 : Jugador, p2 : Jugador) : void

- mostrarTauler(blanc : Jugador<Pieza>, negre : Jugador<Pieza>) : void

- iniciarJuegoBlancas() : ArrayList<Pieza>

- iniciarJuegoNegras() : ArrayList<Pieza>

- **mostrarMenu:** Mostra el menú principal i permet a l'usuari seleccionar entre jugar una partida nova, reproduir una partida des d'un fitxer o sortir del programa.
- **jugarNovaPartida:** Inicia una nova partida d'escacs, permet a dos jugadors introduir els seus torns de forma repetitiva, es mostra el tauler en cada moviment. Al finalitzar el joc es guarda la partida en un fitxer al final.
- **reproduirPartida:** Carrega una partida guardada des d'un fitxer i la reproduceix torn a torn, mostrant el tauler després de cada moviment.
- **llegirTorns:** Demana a l'usuari el nom d'un fitxer per carregar una partida i retorna la llista de torns guardada en el fitxer especificat. Si el fitxer no existeix o hi ha un error en la lectura, es captura l'excepció i s'invoca el mètode de forma recursiva.

- **tornToPosition:** Converteix un torn introduït pel jugador (per exemple, "E4 F3") en coordenades del tauler, mou la peça corresponent del jugador p1 i elimina la peça capturada del jugador p2.
- **mostrarTauler:** Crea un tauler de tipus char de mida 8x8 per mostra l'estat actual del tauler d'escacs amb les peces dels jugadors blanc i negre, representant-les visualment.
- **iniciarJuegoBlancas:** Inicialitza les peces blanques en les seves posicions inicials en un tauler d'escacs.
- **iniciarJuegoNegras:** Inicialitza les peces negres en les seves posicions inicials en un tauler d'escacs.