Programació Avançada

Pràctica 2. Curs 2025/2026



Centre universitari adscrit a la





Continguts de la presentació

- 1. Temàtica de la pràctica
- 2. Objectiu de l'activitat
- 3. Classe InventariJugador
- 4. Classe SequenciaOrdenada<E>





Títol: Gestió d'inventaris amb seqüències enllaçades

En aquesta pràctica treballarem amb dues classes fonamentals:

- InventariJugador: implementa una seqüència enllaçada circular amb capçalera.
 - representa l'inventari d'un jugador, el jugador és la capçalera, la resta d'elements són els articles del seu inventari.
- SequenciaOrdenada<E>: implementa una seqüència enllaçada lineal ordenada sense capçalera d'elements genèrics.
 - representa una seqüència de tots els jugador i els seus inventaris (ordenats pels noms dels jugadors).
 - també representa una seqüència de tots els articles de la botiga (ordenats per preu).





Objectiu de l'activitat

L'objectiu és entendre el funcionament de les seqüències enllaçades.

Treballar dues seqüències: una lineal i una altra circular.

Com treballar una sequència ordenada i com mantenir l'ordre dels elements de manera eficient.

Implementar tant mètodes a la classe original, com a la classe Node, alguns seran recursius.

Implementar la interfície Cloneable i la interfície Iterable.

Utilitzar estructures genèriques <E>.

Utilització d'una classe MyException pròpia.





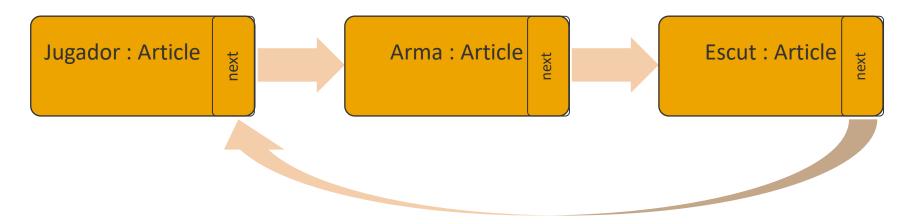
Classe InventariJugador

Un inventari personalitzat per a cada jugador representa una seqüència d'Articles.

Capçalera: és una instància de la classe Article amb la informació del jugador

- Nom del jugador.
- Les seves monedes disponibles.

Una seqüència circular d'articles (el primer element és la capçalera i l'últim element apunta a la capçalera).







Classe InventariJugador

```
InventariJugador
- cap : Node
+ InventariJugador(nomJugador : String, monedesJugador : int)
                                                                                       Node
+ getNomJugador() : String
+ getMonedesJugador() : int
+ setMonedesJugador(monedesJugador : int) : void
                                                                                       ~ sequent : Node
+ afegirFinal(article : Article) : void
                                                                                       ~ info : Article
+ afegirPrincipi(article : Article) : void
+ eliminar(articleBuscat : Article) : boolean
                                                                                       ~ Node(info : Article)
+ getTres(index : int) : ArrayList<Article>
+ qetSize() : int
+ esBuit() : boolean
+ toString() : String
+ compareTo(altre : InventariJugador) : int
                                                                     Article
                                                                     - tipus : ArticleTipus {readOnly}
                                                                     - preu : int
                              <<enumeration>> ArticleTipus
                                                                     - nom : String {readOnly}
                              JUGADOR
                                                                     + Article(nomJugador : String)
                              RECURS
                                                                     + Article(nom : String, preu : int, tipus : ArticleTipus)
                              POCIO
                                                                     + getNom() : String
                              ARMADURA
                                                                     + getPreu() : int
                              ARMA
                                                                     + setPreu(monedesJugador : int) : void
                                                                     + compareTo(other : Article) : int
                                                                     + equals(obj : Object) : boolean
                                                                     + hashCode() : int
                                                                     + toString() : String
```





Centre universitari adscrit a la

Classe SequenciaOrdenada<E>

Una llista enllaçada ordenada i genèrica.

Implementa una seqüència enllaçada simple sense capçalera.

Manté els elements ordenats segons el mètode compareTo() de la classe genèrica E.

No admet elements repetits.

Implementa Iterable $\langle E \rangle \rightarrow$ permet fer servir **foreach** directament.

També disposa d'un mètode clonar() per duplicar la llista.

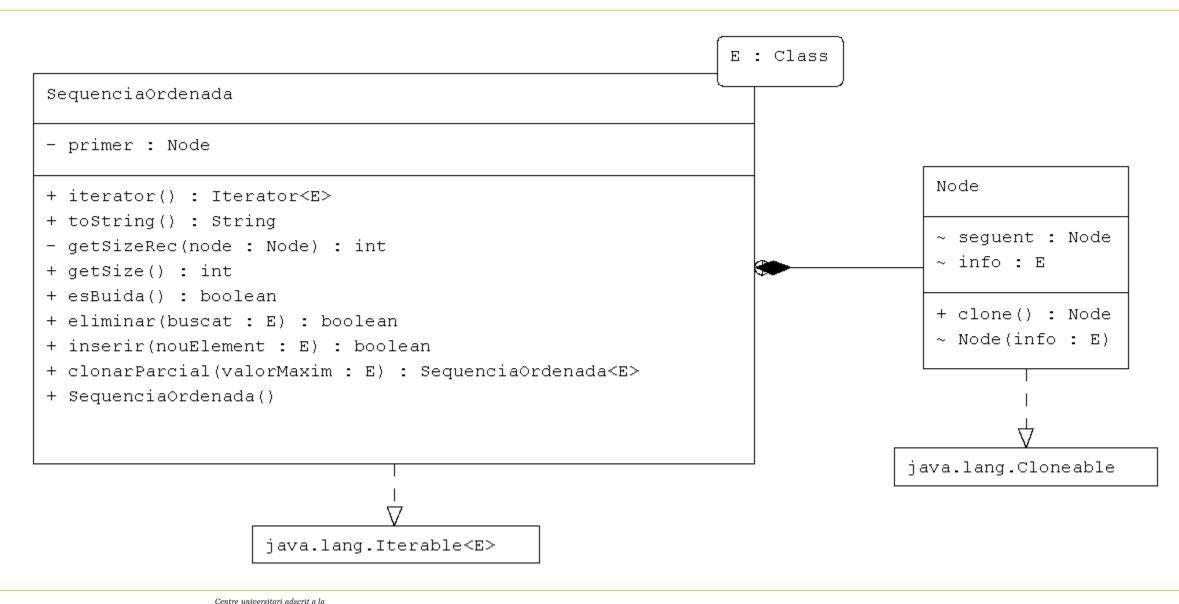
Ús:

- SequenciaOrdenada<InventariJugador> jugadors;
- SequenciaOrdenada<Article> botiga;





Classe SequenciaOrdenada<E>







Recordeu:

- Les pràctiques no són recuperables, ni es poden convalidar d'un curs anterior.
- Les pràctiques s'han de treballar i lliurar amb GITHUB, cada alumne com a mínim ha de fer 3 commits.

Activitat	Avaluació	Data límit
Pràctica 1: Col·leccions genèriques i fitxers	3%	Dilluns 6 d'octubre a les 23 hores
Pràctica 2: Seqüències enllaçades	6%	Dilluns 20 d'octubre a les 23 hores
Pràctica 3: Arbres binaris	6%	Dilluns 3 de novembre a les 23 hores
Pràctica 4: Arbres ACB i tècniques de disseny	6%	Dilluns 17 de novembre a les 23 hores
Pràctica 5: Tècniques de disseny i Backtracking	9%	Dijous 11 de desembre a les 23 hores



