

Gestió d'un hospital

Professors d'IIP - DSIC - UPV

Es desitja realitzar una aplicació per gestionar l'ingrés i l'alta de pacients d'un hospital. Per facilitar el desenvolupament d'aquest exercici, es suposen disponibles (en PoliformaT):

- La classe `GestorHospital`, una classe programa que permet realitzar l'ingrés, l'alta de pacients i la consulta de les dades de tots els pacients i dels pacients crítics d'un hospital.
- La classe `Pacient`, una classe tipus de dades que permet representar un pacient mitjançant els atributs: `nif` (`String`), `nom` (`String`), `edat` (`enter`), `estat` (`enter` que indica l'estat de gravetat del pacient, i que pot prendre els següents valors: 1 per estat crític, 2 per a greu, 3 per a moderat, 4 per a lleu i 5 si està sa). A més, estan definides les constants públiques estàtiques: `CRITIC` = 1, `GREU` = 2, `MODERAT` = 3, `LLEU` = 4 i `SA` = 5, representant els estats de gravetat possibles d'un pacient.

D'aquesta classe s'utilitzaran les constants i els mètodes públics que apareixen en el resum de la seua documentació que es mostra a continuació.

Field Summary

Fields

Modifier and Type	Field and Description
static int	<code>CRITIC</code> Constant que representa l'estat crític.
static int	<code>GREU</code> Constant que representa l'estat greu.
static int	<code>LLEU</code> Constant que representa l'estat lleu.
static int	<code>MODERAT</code> Constant que representa l'estat moderat.
static int	<code>SA</code> Constant que representa l'estat sa.

Constructor Summary

Constructors

Constructor and Description
<code>Pacient(java.lang.String nif, java.lang.String nom, int edat)</code> Constructor d'un objecte <code>Pacient</code> amb NIF <code>nif</code> , de nom <code>nom</code> , de edat anys i estat és un valor aleatori en l'interval <code>[CRITIC,LLEU]</code> .

Method Summary

Methods

Modifier and Type	Method and Description
int	<code>getEdat()</code> Consultor que torna l'edat del pacient actual.
int	<code>getEstat()</code> Consultor que torna l'estat del pacient actual.
java.lang.String	<code>getNif()</code> Consultor que torna el NIF del pacient actual.
java.lang.String	<code>getNom()</code> Consultor que torna el nom del pacient actual.
java.lang.String	<code>toString()</code> Torna un <code>String</code> amb la informació del pacient actual amb el següent format: "nom nif edat est" on est és "CRITIC", "GREU", "MODERAT", "LLEU" o "SA" en funció de l'estat del pacient (per exemple, "Pepe Pérez Santarosa 74747474 46 LLEU").
void	<code>tractament()</code> Modificador que incrementa en uno l'estat del pacient actual.

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `wait`, `wait`, `wait`

En la figura 1 es mostra la representació gràfica d'un objecte de la classe `Pacient`.

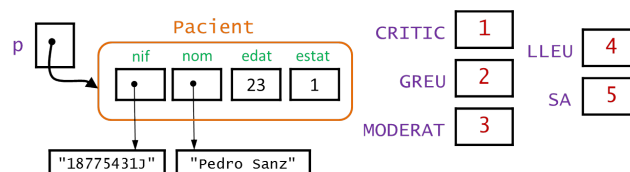


Figura 1: Representació gràfica d'un objecte `Pacient`.

Es demana: completar la classe que segueix, tenint en compte que els seus atributs seran privats i els seus mètodes públics o privats i només els que s'indiquen a la classe.

- La classe `Hospital` conté la informació dels pacients que ocupen els llits d'un hospital. Un `Hospital` té un número màxim de llits `MAX_LLITS = 200` i per representar-los s'utilitza un array `llits` d'objectes de tipus `Pacient` junt amb un atribut `lliures` que indica el número de llits lliures de l'hospital en un moment donat ($0 \leq \text{lliures} \leq \text{MAX_LLITS}$). El número de cada llit coincideix amb la seua posició a l'array de pacients, de manera que `llits[i]` és el `Pacient` que ocupa el llit `i` o és `null` si el llit està lliure (la posició 0 no s'utilitzarà). També té un atribut `critics` ($0 \leq \text{critics} \leq \text{MAX_LLITS}$) que indica el número de pacients de l'hospital en estat crític (és a dir, aquells pacients amb estat igual a `CRITIC`). A més per emmagatzemar l'històric de pacients que han visitat l'hospital, s'utilitza un array `historic` amb els NIF dels pacients junt amb un atribut `pacients` que indica el número de pacients que han passat per l'hospital ($0 \leq \text{pacients} \leq \text{MAX_PACIENTS}$ on `MAX_PACIENTS = 5000` és el número màxim de pacients considerat).

En la figura 2 es mostra la representació gràfica d'un objecte de la classe `Hospital`.

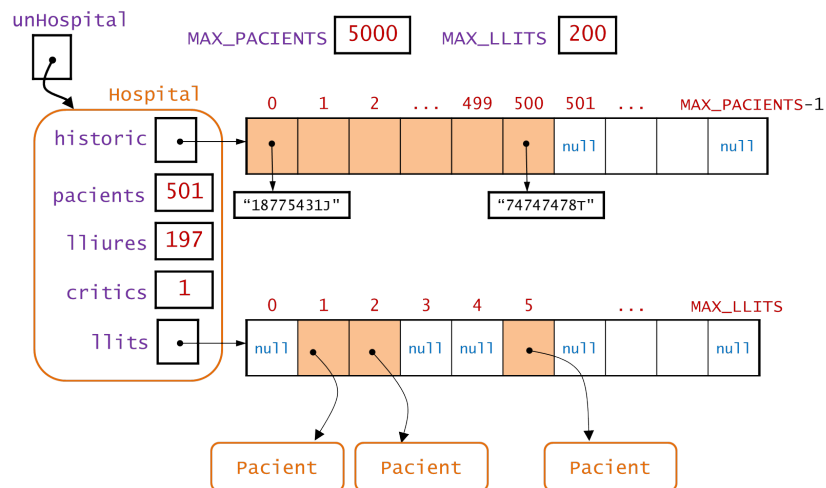


Figura 2: Representació gràfica d'un objecte `Hospital`.

Els mètodes d'aquesta classe són:

- `public Hospital()`. Constructor d'un hospital. Quan es crea un hospital, tots els llits estan lliures, no hi ha pacients i, per tant, no hi ha pacients crítics ni històric de NIF.
- `public int getLliures()`. Consultor del número de llits lliures.
- `public int getPacientsHistoric()`. Consultor del número de pacients en l'històric.
- `public int getCritics()`. Consultor del número de llits ocupats per pacients en estat crític.
- `public boolean hiHaLliures()`. Torna `true` si a l'hospital hi ha llits lliures i torna `false` en cas contrari.
- `public int primerLliure()`. Torna el número del primer llit lliure (el de número menor) si hi ha llits lliures o torna `-1` si no hi ha.
- `public boolean ingresarPacient(String nif, String nom, int edat)`. Si hi ha llits lliures, el primer d'ells (el de número menor) passa a estar ocupat pel pacient amb NIF `nif`, de nom `nom` i edat `edat`. Si el pacient no està en l'històric, s'afegeix. Torna `true` si s'ha fet l'ingrés o torna `false` en cas contrari (si no hi ha llits lliures).
- `private int cercar(String nif)`. Donat un NIF `nif`, comprova si està a l'històric de NIF de l'hospital. Si està, torna la posició que ocupa a l'array `historic`. En cas contrari, torna `-1` indicant que no s'ha trobat.
- `private void afegirHistoric(Pacient p)`. Donat un pacient `p`, si el seu NIF no està en l'històric, l'afegeix.
- `public Pacient getPacient(int i)`. Donat un número de llit vàlid ($1 \leq i \leq \text{MAX_LLITS}$), torna el pacient que ocupa el llit `i` de l'hospital o torna `null` si el llit està lliure.
- `public void donarAltes()`. Es suministra el tractament (mètode `tractament()` de `Pacient`) a tots els pacients de l'hospital i a aquells pacients sans (l'estat dels quals és `SA`) se'ls dona l'alta mèdica (s'invoca a `donarAltaPacient(int)`).
- `private void donarAltaPacient(int i)`. Donat un número de llit vàlid ($1 \leq i \leq \text{MAX_LLITS}$), el llit `i` de l'hospital passa a estar lliure.

- `public String toString()`. Sobreescritura del mètode `toString()` d'`Object` que torna un `String` amb la informació de tots els llits de l'hospital actual o "Hospital buit" si no hi ha pacients a l'hospital. Per exemple,

```

1      María Medina Muñoz    85859463H    30    LLEU
2      Pepe Pérez Santarosa  74747474I    46    GREU
3      lliure
4      Juan López Ayala      12345678J    50    CRITIC
5      lliure
...
200    Andrés Sánchez Enol   98765432M    29    MODERAT

```

- `public int[] pacientesCritics()`. Torna un array amb els números de llit ocupats pels pacients crítics (és a dir, aquells pacients amb estat igual a `CRITIC`). L'array que s'ha de tornar tindrà tants elements com pacients crítics hi ha a l'hospital. Si l'hospital està buit, l'array tindrà 0 elements.

El resum de la seua documentació (constants i mètodes públics) es mostra a continuació.

Field Summary

Fields

Modifier and Type	Field and Description
static int	<code>MAX_LLITS</code> Constant que representa el número màxim de llits de qualsevol hospital.
static int	<code>MAX_PACIENTS</code> Constant que representa el número màxim de pacients que poden passar per qualsevol hospital.

Constructor Summary

Constructors

Constructor and Description
<code>Hospital()</code> Constructor d'un objecte <code>Hospital</code> .

Method Summary

Methods

Modifier and Type	Method and Description
void	<code>donarAltes()</code> Es suministra el tractament() a tots els pacients de l'hospital i a aquells pacients sans (l'estat dels quals és 5) se'ls dona l'alta mèdica (s'invoca a <code>donarAltaPacient(int)</code>).
int	<code>getCritics()</code> Consultor del número de llits ocupats per pacients crítics a l'hospital actual.
int	<code>getLliures()</code> Consultor del número de llits lliures de l'hospital actual.
Pacient	<code>getPacient(int i)</code> Torna el <code>Pacient</code> que ocupa el llit <code>i</code> de l'hospital actual o null si el llit està lliure.
int	<code>getPacientsHistoric()</code> Consultor del número de pacients en l'històric de l'hospital actual.
boolean	<code>hiHaLliures()</code> Consultor que torna true si a l'hospital hi ha llits lliures i torna false en cas contrari.
boolean	<code>ingresarPacient(java.lang.String nif, java.lang.String nom, int edat)</code> Si hi ha llits lliures, el primer d'ells (el de número menor) passa a estar ocupat pel pacient de nif <code>nif</code> , nom <code>nom</code> i edat <code>edat</code> .
int[]	<code>pacientesCritics()</code> Torna un array amb els números de llit ocupats pels pacients crítics (és a dir, aquells pacients amb estat igual a 1).
int	<code>primerLliure()</code> Consultor del número del primer llit lliure de l'hospital si hi ha llits lliures o torna un -1 si no hi ha.
java.lang.String	<code>toString()</code> Sobreescritura del mètode <code>toString()</code> de <code>Object</code> .

Methods inherited from class java.lang.Object

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `wait`, `wait`, `wait`