

Nombre: _____

R.U.N: _____

Paradigmas de la Programación

Evaluación Integradora

Docente: Aaron Pavez G.

26 de marzo 2020

Instrucciones:

1. Lea cuidadosamente cada pregunta antes de responder.
2. Escriba su nombre en cada hoja de respuesta.
3. Utilice solamente las hojas de respuestas.
4. Para cualquier implementación de código, utilice el lenguaje de programación Java.

Nombre: _____

R.U.N: _____

Ejercicio 1

El siguiente código detiene su ejecución por un error en tiempo de ejecución. Implemente el código necesario alrededor de "n" para que la ejecución termine exitosamente, sin errores.

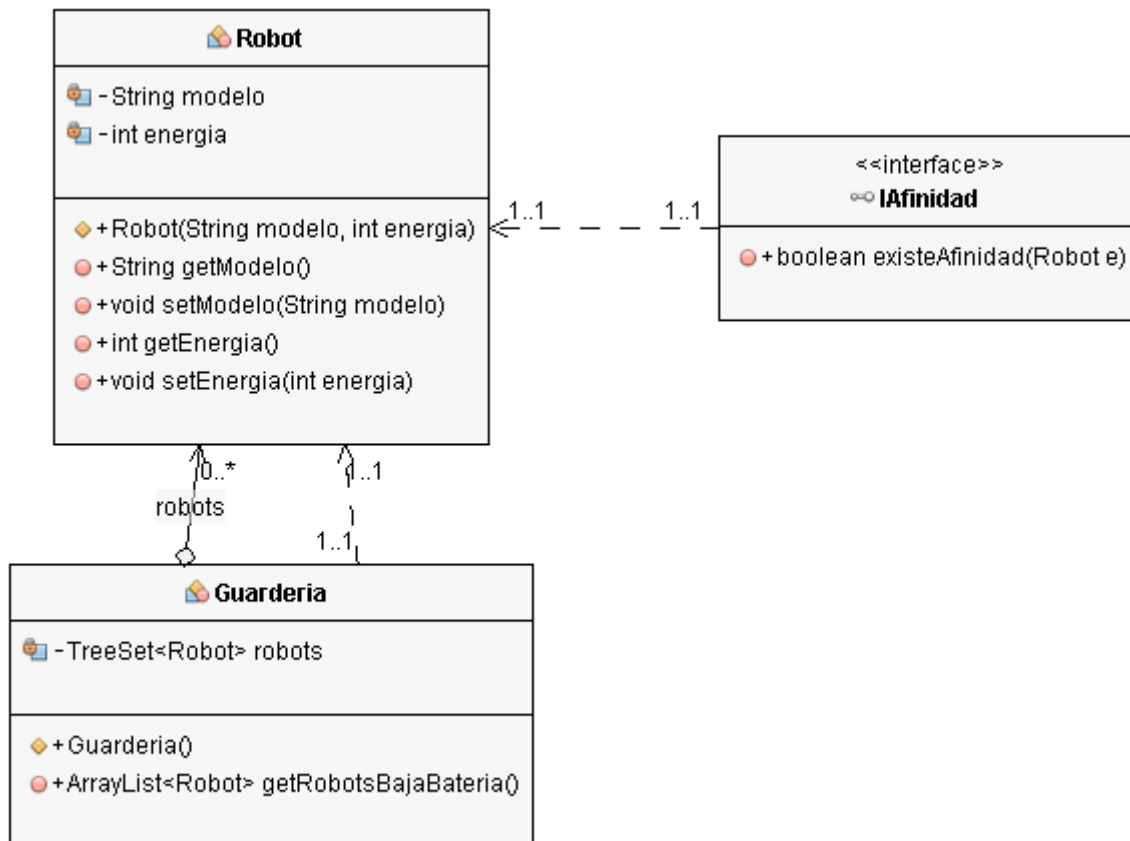
Restricción: Crear una clase que pueda ser utilizada en un try/catch y permita controlar una excepción de tipo `ArithmeticException`.

Escriba toda la clase

(0.5Pts)

Ejercicio 2

Una guardería de Robots, necesita de su ayuda ya que nada funciona como debería.



Nombre: _____

R.U.N: _____

a) El sistema de orden no funciona.

Observación: Una guardería almacena desde 0 a N instancias de Robots.

Para que el sistema ordene correctamente, ¿En qué clase se deberá aplicar la interfaz Comparable?

R: _____ (0.5Pts)

Implemente el método compareTo con su lógica interna.

Observación: Se necesita que se ordene de forma ascendente de acuerdo con su Modelo y por cada grupo de Modelo se deberá ordenar por la carga de Energía más alta a la más Baja (Descendente).

Sin Orden		Con Orden	
Modelo	Energía	Modelo	Energía
B	15	A	10
A	5	A	5
B	1	B	15
A	10	B	1

(1Pts)

Nombre: _____

R.U.N: _____

- b) Bajo la implementación pedida en el punto “a”, ¿Qué sucedería si se ingresa una nueva instancia, pero ya se cuenta con una instancia dentro del TreeSet con los mismos valores en los atributos?

¿Qué sucede?	(0.25Pts)
¿Por qué?	(0.25Pts)

- c) Implemente el método getRobotsBateriaBaja de la Clase Guarderia.

Observación: Retorne una lista con todos los robots que tengan energía inferior a 30.

```
public ArrayList<Robot> getRobotsBajaBateria() { (1Pts)
```

```
}
```

Nombre: _____

R.U.N: _____

d) **Escriba la interfaz** IAfinidad. (Escribir Interfaz, no implementar)

(0.5Pts)

Nombre: _____

R.U.N: _____

- e) Según el código interno, los Robots solamente tienen afinidad con los Modelos que se listan antes a él en el TreeSet. (Asuma que tiene correctamente implementado compareTo)

Con Orden	
Modelo	Energía
A	10
A	5
B	15
B	1

Ejemplo:

El Modelo B con Energía 15, tiene afinidad con los robots A -5 y A-10.

El Modelo A con energía 5, solamente tiene afinidad con A-10.

Implemente el método existeAfinidad.

```
public boolean existeAfinidad(Robot e){
```

(0.5Pts)

```
}
```


Nombre: _____

R.U.N: _____

Implemente el método que retorna una lista de robot afines a "Robot r".

```
public ArrayList<Robot> getListaAfinidadPara(Robot r){ (1Pts)
```

```
}
```