

ESCUELA COLOMBIANA DE INGENIERÍA

DESARROLLO ORIENTADO POR OBJETOS [DOPO-POOB]

HERENCIA - EXCEPCIONES

S07- S09 : 2025-2

Memoria

Presente el mapa de memoria correspondiente a:

- En el juego se tienen dos flotas “LA GRAN FLOTA BLANCA” y “LA GRAN ARMADA DE CASTILLA”
- “LA GRAN ARMADA DE CASTILLA” tienen un barco (número 900); un portaaviones (identificado con 333) y un avión (identificado con 'chkgood'). El avión corresponde al portaaviones 333 y actualmente está en el aire.

Algunos métodos. Diseñando y codificando

Implementen algunos de los siguientes métodos de la clase `Fleet`

Method Summary	
void	moveNorth() Moves the fleet one position north. The world board is circular. longitude [0,180] latitude [-90,90]. Coordinates.
void	advance(int dLon, int dLat) Moves all machines the defined distance. The world board is circular. longitude[0,180] latitude [-90,90]. Parameters: dLon - advance in longitude dLat - advance in latitude
<code>ArrayList<Machine></code>	willBeDestroyed(int longitude, int latitude) Consults the machines that may be affected by an explosion at the given position. Many machines can be at one coordinate. Airplanes in the air are not destroyed.
<code>ArrayList<Machine></code>	weakMachines() Consults the weak machines of a fleet. A ship is weak if it has fewer than five sailors; an airplane, if it has no main pilot; and an aircraft carrier, if it is a weak ship or any of its airplanes in the air is weak. Returns: The weak machines
public boolean	isGoodAttack(int longitude, int latitude) Checks if an attack location is suitable (destroys enemy elements without causing own casualties). Parameters: longitude - longitude of the explosion latitude - latitude of the explosion
void	attack(int lon, int lat) Moves all machines that are not weak towards the indicated attack position (lon, lat). Parameters: lon - longitude lat - latitude

Nueva máquina. Diseñando y codificando.

Las flotas ahora van a tener cápsulas submarinas, sin tripulantes, que piden instrucción a una máquina nodriza. Las máquinas nodrizas pueden ser un barco u otra cápsula. Las capsulas alcanzan profundidades superiores a 8.000 mts que las hace inmunes a ataques y NUNCA son débiles.

- Presentar el diseño estructural
- Analizar si debe modificar el diseño de los métodos desarrollados.
- Diseñar los métodos que requieren cambios.
- Realizar la codificación completa de esta nueva clase.

Nuevo comportamiento. Diseñando y codificando

Los marinos, aviones y cápsulas están preparados para auto-destruirse. Los marinos y barcos se auto-destruyen si reciben la instrucción. Las cápsulas se auto-destruyen si su nodriza es destruida. Todos deben poder informar la causa por la que tomaron la decisión. Las flotas conocen todos los elementos que se han auto destruido.

- Presentar el diseño estructural
- Realizar la codificación completa del nuevo elemento.

Otros métodos. Diseñando y codificando.

Implementen los métodos de la clase `Fleety Board`

Method Summary Fleet	
<code>void</code>	<code>moveNorth()</code> throws <code>BattleShipException</code> Moves the fleet one position north. If any machine cannot move, the movement of the entire fleet stops. Throws: <code>BattleShipException</code> - if any of the machines could not move north.
<code>ArrayList<Sailor></code>	<code>pilots()</code> throws <code>BattleShipException</code> Consults the pilots of the fleet. Returns: The pilots assigned to the airplanes of the fleets. Throws: <code>BattleShipException</code> - if a pilot is not a sailor of the fleet, a pilot of an airplane assigned to an aircraft carrier is not a sailor of the aircraft carrier, or a pilot is assigned to more than one airplane.
<code>int</code>	<code>power()</code> throws <code>BattleShipException</code> Consults the power of the fleet. The power is the number of non-weak machines. Returns: The power of the fleet. Throws: <code>BattleShipException</code> - if there are fewer sailors than machines in the fleet.

Method Summary Board	
<code>int</code>	<code>toNorth()</code> Consults the number of fleets that completed a full movement
<code>ArrayList<Fleet></code>	<code>infiltrated()</code> throws <code>BattleShipException</code> Consults the fleets that have infiltrated pilots. Returns: The fleets with infiltrated pilots Throws: <code>BattleShipException</code> - if any fleet does not have assigned sailors.
<code>int</code>	<code>power()</code> throws <code>BattleShipException</code> Consults the power of the board. The power is the sum of the power of the fleets. Returns: The power of the board. Throws: <code>BattleShipException</code> - if more than half of the fleets have power issues.

