

Apellido y nombre: _____ Legajo: _____

PRIMER EXAMEN PARCIAL**30/06/2023**

Resolvé el siguiente ejercicio utilizando el lenguaje **Java** y el paradigma orientado a objetos. Lee el enunciado al menos dos veces antes de intentar confeccionar la solución.

ENUNCIADO

La estación espacial Atlantis dispone de puntos de atraque que alquila a naves espaciales de distinto tipo. Cada uno de estos puntos de atraque puede “enganchar” solo una nave, de las cuales por ahora hay dos tipos principales:

- Las naves de tipo Cargo, dedicadas al transporte de mercadería, guardan la información de su carga: descripción y volumen en metros cuadrados.
- Las naves de tipo Cruiser, guardan la cantidad de pasajeros que transportan actualmente.

Todas las naves deben poder informar un “manifiesto”, que en todos los casos consta de datos que hay guardados para todas las naves:

- El nombre del planeta de origen de la nave.
- La cantidad de miembros de la tripulación.

Cada tipo de nave agrega datos particulares al manifiesto:

- Cargo: el detalle de la carga con formato: "**descripcion (volumen m3)**".
- Cruiser: El porcentaje de pasajeros, respecto al total (pasajeros + tripulantes).

A partir del modelo enunciado, se pide:

- 1) Desarrollar un método llamado **mostrarManifiestos()** que muestre por consola los manifiestos de todas las naves que ya haya estacionadas.
- 2) Desarrollar un método llamado **estacionarNave()** que reciba una nave y la intente estacionar en el primer punto de atraque libre, siempre que la nave esté en condiciones de estacionarse:
 - Las naves Cargo pueden estacionarse si el volumen de su carga no supera 100 m3.
 - Las naves Cruiser pueden estacionarse si y sólo si hay pasajeros.
- 3) Desarrollar un test en el método **main** del proyecto que resuelva lo siguiente:
 - a. Crear la estación espacial con 5 puntos de atraque (inicialmente libres)
 - b. Intentar estacionar las siguientes naves:

Tipo de nave	Planeta de origen	Tripulantes	Carga	Cantidad de pasajeros
Cargo	Neptuno	21	Baterías recargables 120m ³	<i>No aplica</i>

Cruiser	Mercurio	8	No aplica	0
Cargo	Marte	15	Sustancias cósmicas 90m ³	No aplica
Cruiser	Urano	48	No aplica	12

c. Mostrar los manifiestos de las naves que se hayan podido estacionar.

FORMATO DE ENTREGA

Archivo .zip o .rar nombrado de la siguiente manera (reemplazando lo que está en verde):

INSPT-ProgII-p1-2023-EstacionEspacial-TUPELLIDO-TUNOMBRE

El .zip o .rar debe contener el proyecto Java, el cual debe tener la misma nomenclatura.

No cumplir con alguna indicación del formato de entrega disminuye la nota del examen.

CRITERIOS DE EVALUACIÓN

Para considerar aprobado el examen, el mismo debe demostrar la correcta aplicación de los siguientes conceptos de la programación orientada a objetos y el lenguaje Java:

- Definición de clases y asignación adecuada de sus responsabilidades.
- Encapsulamiento, ocultando detalles de implementación y utilizando métodos getters y setters sólo cuando corresponda.
- Modularización reusable y mantenible, usando funciones con correcta parametrización, aplicando alta cohesión y bajo acoplamiento.
- Correcta implementación de los constructores.
- Validación de los datos que ingresan al sistema.
- Aplicación de herencia y polimorfismo, incluyendo interfaces.
- Aplicación de los principios S.O.L.I.D
- Algoritmos precisos, finitos y eficientes.
- Buenas prácticas de programación:
 - Uso de constantes en lugar de números mágicos.
 - Nombres de variables, métodos y clases descriptivos, que sigan la convención del lenguaje.
 - Uso adecuado de los ciclos. No alterar la naturaleza de los mismos utilizando, por ejemplo, instrucciones de control como **break** y **continue**.
 - Código ordenado e indentado correctamente.

% Correcto	0 a 20	25 a 45	50 a 55	60	65 a 70	75	80	85 a 90	95	100
Nota	1	2	¿4?	4	5	6	7	8	9	10

(Escala solo con fines orientativos)