

Primer Parcial

Tema 1

Objetivo

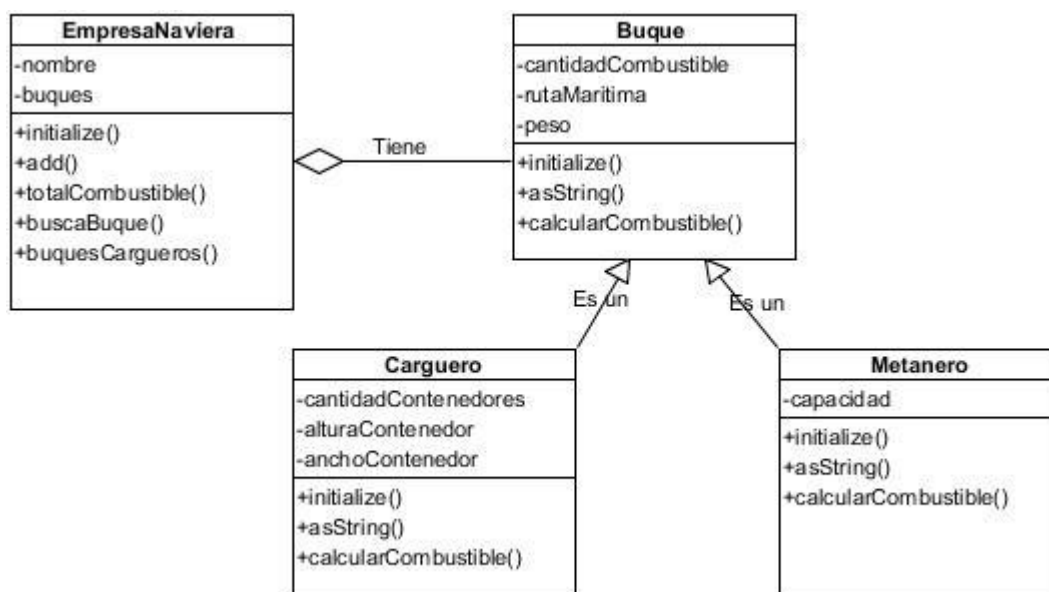
Evaluar al estudiante en la parte práctica de la unidad nro. 4, Paradigma Orientado a Objetos, de los siguientes temas: Clases, objetos, paso de mensajes, composición, herencia, polimorfismo y colecciones en Smalltalk.

Condiciones de trabajo:

- El alumno deberá desarrollar un programa, utilizando el correspondiente entorno.
- El código fuente del alumno deberá quedar guardado en la unidad D: nominado con su legajo y nombre, de la siguiente forma: Legajo_ApellidoNombre.st.
- **Es responsabilidad de cada alumno ir guardando periódicamente su archivo, como así también del contenido de los mismos, teniendo la precaución de guardarlo en el disco D: para su posterior backup.**
- En caso de que máquina no funcione correctamente durante el transcurso de la evaluación, debe notificar de esta situación a cualquier docente de la mesa examinadora.
- En ningún caso debe reiniciar la máquina, ya que perderá la totalidad del examen.
- **Es responsabilidad de cada estudiante el contenido que quede guardado al momento de finalizar la evaluación, en todos sus archivos generados durante el examen.**
- El **tiempo previsto** para la realización de este examen es de **1:30 hs.**

Enunciado:

Una empresa naviera requiere desarrollar un software que gestione los buques que utiliza regularmente. Para ello se necesita implementar en Smalltalk un conjunto de clases, obtenidas en la etapa de diseño, que se grafican y detallan a continuación:



Buque, que posee:

Atributos:

- Cantidad de combustible por milla.
- Ruta marítima: cantidad de millas a recorrer para llegar a destino.
- Peso del buque vacío.

Métodos:

1. Inicializador.
2. Métodos de acceso y de modificación.
3. `asString`: que retorna todos los datos de la clase concatenados en un objeto String.
4. `calcularCombustible`: que determina la cantidad de combustible que necesita el buque para llegar a destino estando vacío: cantidad de combustible por milla multiplicado por ruta marítima y multiplicado por el peso del buque cuando esta vacío.

Carguero, es un buque que posee:

Atributos:

- cantidad de contenedores que puede transportar.
- altura del contenedor: todos los contenedores tienen la misma altura.
- ancho del contenedor: todos los contenedores tienen el mismo ancho.

Métodos:

1. Inicializador.
2. Métodos de acceso y de modificación.
3. asString: que retorna todos los datos de la clase, concatenados en un objeto String.
4. calcularCombustible: el cual se obtiene multiplicando la cantidad de combustible que requiere el buque cuando está vacío por la capacidad del carguero, ésta se calcula como la cantidad de contenedores multiplicada por la superficie de un contenedor (altura multiplicada por ancho).

Metanero, es un buque que transporta gas licuado y posee.

Atributos:

- capacidad: cantidad máxima de metros cúbicos de gas licuado que puede transportar el buque.

Métodos:

1. Inicializador.
2. Métodos de acceso y de modificación.
3. asString: que retorna todos los datos de la clase concatenados en un objeto String
4. calcularCombustible: si la capacidad del metanero es mayor a 100 metros cúbicos, la cantidad de combustible es igual a la cantidad de combustible que requiere el buque cuando está vacío más el 60% de dicha cantidad de combustible. Si la capacidad del metanero es menor o igual a 100 metros cúbicos, la cantidad de combustible es igual a la cantidad de combustible que requiere el buque cuando está vacío más el 40 % de dicha cantidad.

EmpresaNaviera: posee:

Atributo:

- nombre de la empresa
- buques (una colección dinámica de buques de tipo Carguero o Metanero).

Métodos:

1. Inicializador.
2. Métodos de acceso y de modificación.
3. add: agrega un buque a la colección.
4. totalCombustible: Mostrar el total de combustible consumido por todos los buques de la empresa.
5. buscaBuque: Mostrar el primer buque que tenga un peso igual a uno recibido como colaborador externo.
6. buquesCargueros: Mostrar los buques Cargueros en los que su cantidad de contenedores no superen una cantidad recibida como colaborador externo.

Requerimientos del parcial:

1. Con lo expuesto anteriormente, usted deberá implementar las clases con sus atributos y métodos mencionados, solo podrá agregar nuevos métodos si lo considera necesario.
2. Podrá probar sus clases, instanciando los objetos en una ventana de workspace sugerida en el archivo primerParcial.ws, la cual podrá usar y modificar según su criterio.

Tabla de valoración de los ítems evaluados

Ítems a Evaluar	Pje	Observaciones	Obtenido
Implementación de la jerarquía de clases, con métodos comunes.	20		
Definición de métodos polimórficos en la jerarquía (calcularCombustible). 3 en total.	15		
Implementación de métodos comunes en la clase EmpresaNaviera (Inicializador. Acceso, modificación y add).	15		
Implementación del método 4 en EmpresaNaviera.	15		
Implementación del método 5 en EmpresaNaviera.	15		
Implementación del método 6 en EmpresaNaviera.	20		
Total	100		