

Alumno	Entrega	Compila	Nota	Visto	Observaciones
					Vehículo no lleva atributo tamaño La propiedad Tamaño debe ser abstracta Mostrar() retorna this.ToString()? Taller: ToString() sin override
Acuña Zustovich, Alejo Matías	SI	SI	D	SI	
Agüero, Leandro	NO	NO	S/C	NO	
					ETamaño Tamaño no es abstracta, y es recursiva (StackOverflowException) Mostrar() no reutiliza código, y es desprolijo Equals(object obj) ni GetHashCode() no se pide en el diagrama Sedan no reutiliza los constructores Taller() es privado Rehacer, mirar los diagramas, seguir las pautas
Aguirre, Matias	SI	SI	D	SI	operator !=(Vehiculo v1, Vehiculo v2) repite código Sedan no reutiliza sus constructores
Alcaine, Tomas	SI	SI	D	SI	Listar(Taller taller, ETipo tipo) no implementa polimorfismo (motivo de desaprobación)
Alegria, Iautaro	SI	SI	A	SI	operator !=(Vehiculo v1, Vehiculo v2) repite código Sedan no reutiliza los constructores
					int esta = 0; utiliza 0 y 1? Utilizar bool, no int No comentar el código en la misma línea, se hace difícil de leer
Aner Scott, Federico	SI	SI	A	SI	
Antunez, Nahuel	NO	NO	S/C	NO	
Arakaki, Nicolás	SI	SI	A	SI	Sedan no reutiliza los constructores
Argibay, Gaston	SI	SI	A	SI	Sedan no reutiliza los constructores
					Sedan no reutiliza los constructores
Arismnedy Uribe , Herik	SI	SI	A	SI	operator !=(Vehiculo v1, Vehiculo v2) repite código
					operator !=(Vehiculo v1, Vehiculo v2) repite código
Balderrama Rocha, Jhossymar	SI	SI	A	SI	Sedan no reutiliza los constructores
Baldovino, Carlos	NO	NO	S/C	NO	
Barbosa, Lucas	SI	SI	D	SI	Listar(Taller taller, ETipo tipo) no implementa polimorfismo (motivo de desaprobación)
Barrionuevo, Ángel	SI	SI	A	SI	operator !=(Vehiculo v1, Vehiculo v2) repite código
Barrios Crespo, Matias	SI	SI	A	SI	Sedan no reutiliza los constructores
Bentosela, Antonio	NO	NO	S/C	NO	
					Equals(object obj) y GetHashCode() no se piden (ver diagrama) operator != repite código de una forma muy extraña Sedan no reutiliza los constructores
Bernheim, Agustin	SI	SI	A	SI	operator != repite código de una forma muy extraña Sedan no reutiliza los constructores
					operator != repite código de una forma muy extraña Sedan no reutiliza los constructores
Bianchini, Alejo Exequiel	SI	SI	A	SI	operator + repetirá taller.vehiculos.Count == taller.espacioDisponible tantas veces como itere. Mejorar lógica.
Bongioanni, Alejandro	SI	SI	A	SI	operator != repite código Equals(object obj) y GetHashCode() no se piden (ver diagrama)
					operator != repite código Sedan no reutiliza los constructores
Bonomo, Cristian	SI	SI	A	SI	operator + demasiados return, muy complicado

Bruschini, Ariel	SI	SI	A	SI	
					Equals(object obj) y GetHashCode() no se piden (ver diagrama)
					operator != repite código
Bustos Gil, Felipe	SI	SI	A	SI	Sedan no reutiliza los constructores
Cabeza, Leandro	SI	SI	D	SI	No cumple lógica ni salida por pantalla (agrega 7 elementos en un límite de 6)
Caffaro, Facundo Darian	SI	SI	A	SI	
					No comentar código al costado de la línea, es desprolijo y difícil de leer
					operator != repite código
Cantero, Luciano	SI	SI	A	SI	Sedan no reutiliza los constructores
Capozzoli, Paula	NO	NO	S/C	NO	
Carata, Josue	NO	NO	S/C	NO	
Cardozo, Sergio Esteban	SI	SI	A	SI	operator != repite código
Cardullo, Matias	SI	SI	D	SI	No cumple lógica ni salida por pantalla (Elimina un elemento de más)
Carranza, Ezequiel	SI	SI	A	SI	
Cea, Agustín	SI	SI	D	SI	No cumple lógica ni salida por pantalla (no filtra las listas)
Ceriana, Daniel	NO	NO	S/C	NO	
Cerro, Santiago	SI	SI	A	SI	
Chaves Osorio, Juliana	SI	SI	D	SI	No cumple lógica ni salida por pantalla (Elimina un elemento de más)
Chena, Paula	SI	SI	D	SI	No cumple salida por pantalla
					Por qué .ToString() a un string?
					operator ==(Vehiculo v1, Vehiculo v2): los string tienen sobrecargado el operador ==
					Colocar la visibilidad de los atributos y métodos, siempre, por reglas de estilo
Cisa, Nahuel	SI	SI	A	SI	Sedan no reutiliza los constructores
Cisneros, Ludmila	NO	NO	S/C	NO	
					Equals(object obj) y GetHashCode() no se piden (ver diagrama)
Clas, Agustin	SI	SI	A	SI	Contains no compara mediante el == como se pide (funciona por que incumplis lo del Equals que marqué antes)
					Equals(object obj) y GetHashCode() no se piden (ver diagrama)
					Sedan(EMarca marca, string chasis, ConsoleColor color) no asigna tipo por defecto
Clas, Guido	SI	SI	D	SI	Se pide no utilizar lambda o funciones anonimas y resolver los ejercicios con lo visto en clase Exists(k => k == vehiculo)
Coloritto, Gonzalo Miguel	NO	NO	S/C	NO	
					ETipo tipo = ETipo.CuatroPuertas; nunca asignaciones en la declaración del atributo.
					Sedan(EMarca marca, string chasis, ConsoleColor color) no asigna tipo por defecto
Corsaro, Cristian Nahuel	SI	SI	D	SI	No aplica polimorfismo
Costanzo, Abril	SI	SI	A	SI	Sedan no reutiliza los constructores
de la Canal, Juan Ignacio	NO	NO	S/C	NO	
					operator != repite código
De Moraiz , Miguel Alejandro	SI	SI	A	SI	Los constructores de Sedan están mal
de Renzis, Bruno	NO	NO	S/C	NO	El repo publicado ya no existe https://github.com/BrunodeRenzis/TPS-LABORATORIO-2/
de Sierra, Mario Nicolás	NO	NO	S/C	NO	
Delgadillo Ortuño, Mauricio	SI	SI	A	SI	operator != repite código
Delgado, Leandro Emmanuel	NO	NO	S/C	NO	

Delocca, Franco	SI	SI	A	SI	operator != repite código
					Los constructores de Sedan están mal
					Pedimos que no utilicen =>
					operator != repite código
Diaz Giosso, Lucas	SI	SI	A	SI	Sedan no reutiliza los constructores
					Ser prolijo!
Dorso, Juan Martin	SI	SI	A	SI	operator != repite código
					ETamaño Tamaño debía ser abstracta
					operator != repite código de una manera extraña
					Equals(object obj) y GetHashCode() no se piden (ver diagrama)
Dos Santos, juan pablo	SI	SI	D		Sedan no reutiliza los constructores
				SI	Contains no compara mediante el == como se pide (funciona por que incumplis lo del Equals que marqué antes)
Duarte, Maximiliano Javier	NO	NO	S/C	NO	
Enriquez, Ignacio	NO	NO	S/C	NO	
Escubilla Taxis, Alejandra	SI	SI	A		operator == validaciones redundantes
					Equals(object obj) y GetHashCode() no se piden (ver diagrama)
				SI	Sedan no reutiliza los constructores
					operator != repite código
Espinosa, Quimey	SI	SI	A	SI	Sedan no reutiliza los constructores
Facello, Nahuel	NO	NO	S/C	NO	
Fajardo, Sebastián	NO	NO	S/C	NO	
Falcioni, Facundo Gabriel	SI	SI	A		operator != repite código
					Sedan no reutiliza los constructores
					operator + si no hay espacio disponible, ni hagas el foreach
				SI	int flag = 0; utiliza 0 y 1? Utilizar bool, no int
Falcone, Amilcar Facundo	NO	NO	S/C	NO	
Farias Villarreal, Ricardo	NO	NO	S/C	NO	
Fernandez, Agustin	SI	SI	A		operator != repite código
				SI	Sedan no reutiliza los constructores
					operator != repite código
					Sedan no reutiliza los constructores
Fernandez, Javier	SI	SI	A		No asigna el Tipo por defecto
					operator + simplificar lógica
					operator + si no hay espacio disponible, ni hagas el foreach
					No comentar código al costado de la línea, es desprolijo y difícil de leer
Fernandez, Maximiliano	SI	SI	A	SI	Sedan no reutiliza los constructores
Fernández, Mariano	SI	SI	A	SI	operator + si no hay espacio disponible, ni hagas el foreach
Fernandez Heredia, Catalina	NO	NO	S/C	NO	
Fernandez Herenda, Natalia	SI	SI	A		operator != repite código
					Sedan no reutiliza los constructores
				SI	operator +: si el primer IF envolviera a todo el código debajo, podrías poner un return menos

Fernandez Manrique, Kevin	SI	SI	D	SI	Última vez que se corrige un proyecto zippeado
					No cumple lógica ni salida por pantalla (no elimina elementos como corresponde)
Ferrero, Macarena	SI	SI	A	SI	operator != repite código
					No reutiliza los constructores de Sedan
Festinese, guadalupe	SI	SI	A	SI	Mal la lógica de los constructores de Sedan
					operator + si no hay espacio disponible, ni hagas el foreach
Franco Ezequiel, Sagnella	SI	SI	A	SI	Me gusta más tu salida por pantalla que la que di yo. Gracias!
					No reutiliza los constructores de Sedan
Friz, Tomás	SI	SI	A	SI	operator != repite código
					No reutiliza los constructores de Sedan
Gabor , Roberta	SI	SI	A	SI	operator + si no hay espacio disponible, ni hagas el foreach
					operator != repite código
Gaitan, Agustin	SI	SI	A	SI	Equals(object obj) y GetHashCode() no se piden (ver diagrama)
					No reutiliza los constructores de Sedan
Garancini, Franco	SI	SI	A	SI	operator + si no hay espacio disponible, ni hagas el foreach
					operator != repite código
Garcia, Joaquin	NO	NO	S/C	NO	No reutiliza los constructores de Sedan
Gil, Valeria	NO	NO	S/C	NO	
Gonzales Flores, mauricio	SI	SI	D	SI	No cumple lógica ni salida por pantalla
Gonzalez, Lucio Andres	SI	SI	A	SI	
González Pérez, Jose Vicente	NO	NO	S/C	NO	
Gracia, Magali Mailen	SI	SI	A	SI	operator != repite código de forma extraña
					No reutiliza los constructores de Sedan
Gutierrez, Federico	SI	SI	D	SI	No cumple lógica ni salida por pantalla
Gutierrez, Javier	SI	SI	D	SI	No cumple lógica ni salida por pantalla
					operator != repite código
Herrera, Martín	SI	SI	A	SI	No reutiliza los constructores de Sedan
					operator != repite código
Huallpa, Wilson	SI	SI	A	SI	No carga el Tipo por defecto en el construcor
Iacobellis, Lucas	SI	SI	D	SI	No cumple lógica ni salida por pantalla
					No reutiliza los constructores de Sedan
Jaimez, Maria Luana	SI	SI	D	SI	Listar(Taller taller, ETipo tipo) no implementa polimorfismo (motivo de desaprobación)
					operator != repite código
Jara Caceres, Claudia	SI	SI	A	SI	No reutiliza los constructores de Sedan
					Se pide no utilizar =>
Lamas, Juan Pablo	NO	NO	S/C	NO	
Landaeta, Miguel	NO	NO	S/C	NO	
Lavilla, Noelia	SI	SI	D	SI	operator != repite código
					No reutiliza los constructores de Sedan
					Listar(Taller taller, ETipo tipo) no implementa polimorfismo (motivo de desaprobación)

				SI	operator != repite código de forma extraña
Ledesma, Ricardo Sebastian	SI	SI	A	SI	No reutiliza los constructores de Sedan
Lemos, Lautaro Lucas	SI	SI	D	SI	Listar(Taller taller, ETipo tipo) no implementa polimorfismo (motivo de desaprobación)
					No reutiliza los constructores de Sedan
Lenzuen, Maximiliano	SI	SI	A	SI	operator + si no hay espacio disponible, ni hagas el foreach
					No reutiliza los constructores de Sedan
Lezama, Lautaro	SI	SI	A	SI	operator + si no hay espacio disponible, ni hagas el foreach
					operator != repite código
Lopez, Carlos	SI	SI	A	SI	No reutiliza los constructores de Sedan
					No reutiliza los constructores de Sedan
Lospalluto, Sasha Oriana	SI	SI	A	SI	operator + si no hay espacio disponible, ni hagas el foreach
					No reutiliza los constructores de Sedan
Marini, Daniel Julio	SI	SI	A	SI	operator + si no hay espacio disponible, ni hagas el foreach
Masotti, Fernando Javier	SI	SI	A	SI	
					operator != repite código Equals(object obj) y GetHashCode() no se piden (ver diagrama) Se pide no utilizar => No reutiliza los constructores de Sedan
Mazzucchelli Puente, Juan	SI	SI	A	SI	operator + si no hay espacio disponible, ni hagas el foreach
Melgarejo, Lorena	SI	SI	D	SI	No cumple lógica ni salida por pantalla
					Los espacios disponibles disminuyen?? taller.espacioDisponible--;
					operator != repite código
Meli, Mateo	SI	SI	A	SI	No reutiliza los constructores de Sedan vehiculos.Capacity?
					operator != repite código
Mendoza, Maximiliano	SI	SI	A	SI	operator + si no hay espacio disponible, ni hagas el foreach
Mengassini, Alan	SI	SI	A	SI	No reutiliza los constructores de Sedan
Merlini, Maximiliano Gastón	NO	NO	S/C	NO	
					operator != repite código Se pide no utilizar => No reutiliza los constructores de Sedan
Miranda, Cristian Gabriel	SI	SI	A	SI	No reutiliza los constructores de Sedan
Montilla, Vicente	SI	SI	A	SI	No reutiliza los constructores de Sedan
					operator != repite código Se pide no utilizar => No reutiliza los constructores de Sedan
Mosteiro, Santiago	SI	SI	A	SI	No reutiliza los constructores de Sedan
Nogueira, jose maria	NO	NO	S/C	NO	
					operator != repite código
Noguera, Leandro	SI	SI	A	SI	No reutiliza los constructores de Sedan
					operator != repite código
					No aplica polimorfismo en Mostrar()
Nuñez, Joaquin	SI	SI	A	SI	No reutiliza los constructores de Sedan

					Se agregan propiedades que no se piden Equals(object obj) y GetHashCode() no se piden (ver diagrama) Los Mostrar() no se reutilizan
Ojeda, Lisbaldy	SI	SI	A	SI	No reutiliza los constructores de Sedan
Olivera Laura, cristian fermin	SI	SI	A	SI	operator != repite código de una forma muy extraña
Palermo, Juan Gabriel	SI	SI	D	SI	No cumple lógica ni salida por pantalla (Elimina un elemento de más)
					Mostrar() y explicit repiten el mismo código No aplica polimorfismo en Mostrar() ni en Listar(Taller taller, ETipo tipo) No reutiliza los constructores de Sedan
Perez, nicolas eduardo	SI	SI	D	SI	operator + repetirá taller.vehiculos.Count == taller.espacioDisponible tantas veces como itere. Mejorar lógica.
Quevedo, Kevin	NO	NO	S/C	NO	
Quiroz, Esteban Marcelo	SI	SI	D	SI	No cumple lógica ni salida por pantalla, mismos errores que Matías Quiroz. Por confesión de copia vía Slack, pierde la posibilidad de promocionar la materia
Quiroz, Matias Mariano	SI	SI	D	SI	No cumple lógica ni salida por pantalla, mismos errores que Esteban Quiroz.
					Nunca asigna Tipo por defecto No aplica polimorfismo en Mostrar()
Ramirez, jeremias daniel	SI	SI	A	SI	Ser prolijo!
Reynoso, Juan Cruz	NO	NO	S/C	NO	
					Vehiculo operator != no reutiliza el == No aplica polimorfismo en Mostrar()
Rios Paredes, Ricardo Nicolas	SI	SI	A	SI	Nunca asigna Tipo por defecto
					Vehiculo operator != no reutiliza el == No se pide el método Equals() (ver diagrama) operator + no valida el espacio disponible
Rivero, Tomás Agustín	SI	SI	A	SI	No reutiliza los constructores de Sedan
					Por qué castea string a string? (string)p.Mostrar();
Rocha, Francisco	SI	SI	A	SI	No reutiliza los constructores de Sedan
Rodicio, Julian	SI	SI	A	SI	No aplica polimorfismo en Mostrar()
					No aplica polimorfismo en Mostrar()
Rodriguez Ossa, Alexis Kevin	SI	SI	A	SI	No reutiliza los constructores de Sedan int flag = 0; utiliza 0 y 1? Utilizar bool, no int
					Mostrar() y explicit repiten el mismo código No aplica polimorfismo en Mostrar()
Salas, Dario	SI	SI	A	SI	No reutiliza los constructores de Sedan
					Vehiculo operator != no reutiliza el == Reutiliza mal los constructores de Sedan Listar(Taller taller, ETipo tipo) resolviste para este caso, pero ante una modificación (nueva clase que no coincida su nombre con el enumerado) tenés que emparcharlo todo. Ojo!
Sanchez, Matias	SI	SI	A	SI	operator + si no hay espacio disponible, ni hagas el foreach
Sandoval, Matias	SI	SI	D	SI	No cumple lógica ni salida por pantalla (Elimina un elemento de más)

					<p>Última vez que se corrige un repositorio que no cumpla las condiciones</p> <p>ETamano Tamano debía ser abstract</p> <p>Pedimos que no utilicen =></p> <p>Equals(object obj) y GetHashCode() no se piden (ver diagrama)</p> <p>Constructores de Sedan no se reutilizan</p> <p>No aplica polimorfismo en Mostrar()</p> <p>Las entregas deben ser prolijas, sin código comentado y bien indentado</p> <p>int ExisteVehiculo = 0; utiliza 0 y 1? Utilizar bool, no int</p> <p>ExisteVehiculo no cumple reglas de estilo</p> <p>operator + mejorar</p>
Santoliquido, Juan Luca	SI	SI	D	SI	
Sbaglia, Santiago	NO	NO	S/C	NO	
Seif, Mariano	SI	SI	D	SI	<p>Vehiculo operator != no reutiliza el ==</p> <p>protected ETamano tamano; no se pide, leer los diagramas</p> <p>Reutiliza mal los constructores de Sedan</p> <p>Listar debía ser static</p> <p>if(taller.vehiculos.Count > 0) validación innecesaria</p>
					<p>Las entregas deben ser prolijas, no dejar código comentado o mal indentado</p> <p>Vehiculo operator != no reutiliza el ==</p> <p>Constructores de Sedan no se reutilizan</p>
Sellanes, Matias	SI	SI	D	SI	<p>if (v.Tamano == Vehiculo.ETamano.Grande)? No valida por tipo del objeto sino por su tamaño? Si agrego la clase Camión y su tipo es Grande?</p>
					<p>Constructores de Sedan no se reutilizan</p>
Señeriz, Pedro	SI	SI	A	SI	<p>if(taller.vehiculos.Count < 6)? No utiliza atributo espacioDisponible?</p>
Simala, Jean Lucca	NO	NO	S/C	NO	
Solari, Rodolfo	SI	SI	A	SI	<p>Equals(object obj) y GetHashCode() no se piden (ver diagrama)</p> <p>Vehiculo operator != no reutiliza el ==</p> <p>Constructores de Sedan no se reutilizan</p> <p>operator + repetirá taller.vehiculos.Count == taller.espacioDisponible tantas veces como itere. Mejorar lógica.</p>
					<p>ToString(), Equals(object obj) y GetHashCode() no se piden (ver diagrama)</p> <p>Vehiculo operator != no reutiliza el ==</p>
Soñez, Martín Sebastián	SI	SI	A	SI	<p>int flag = 0; utiliza 0 y 1? Utilizar bool, no int</p>
					<p>Equals(object obj) y GetHashCode() no se piden (ver diagrama)</p>
Soria, Federico Martín	SI	SI	A	SI	<p>Constructores de Sedan no se reutilizan</p> <p>int flag = 0; utiliza 0 y 1? Utilizar bool, no int</p>
					<p>Ordenar repositorio</p>
Stemberger, Ivan	SI	SI	D	SI	<p>No cumple lógica ni salida por pantalla (no filtra las listas)</p>
					<p>Vehiculo operator != no reutiliza el ==</p>
Stocki, Alex	SI	SI	A	SI	<p>Constructores de Sedan no se reutilizan</p>
Szuchet, Ezequiel	NO	NO	S/C	NO	
Tapia, Ramiro	NO	NO	S/C	NO	
					<p>operator != repite código</p> <p>Equals(object obj) y GetHashCode() no se piden (ver diagrama)</p>
Tellechea, Leon	SI	SI	A	SI	<p>Constructores de Sedan no se reutilizan</p>

Texis, Federico	NO	NO	S/C	NO	El repositorio debe ser exclusivamente para los TPs, cumpliendo las condiciones de entrega
Tirante, Sergio	SI	SI	A	SI	Constructores de Sedan no se reutilizan
Torrez, Ailen	SI	SI	A	SI	
Troncoso, Elias	SI	SI	A	SI	Vehiculo operator != no reutiliza el ==
Uliczki, Micaela	SI	SI	A	SI	Constructores de Sedan no se reutilizan
Varela Vargas, Leandro Gastón	NO	NO	S/C	NO	
Vazquez, Ezequiel Omar	SI	NO	D	SI	El TP no cuenta con solución
Vila, León	SI	SI	D	SI	operator string(Vehiculo p) debe ser explícito Las entregas deben ser prolijas Sedan: no se reutilizan sus constructores Listar(Taller taller, ETipo tipo) no aplica polimorfismo operator +(Taller taller, Vehiculo vehiculo) la validación de espacio disponible se realizará muchas veces. Mejorar.
Villegas, Alvaro	SI	SI	D	SI	Tamano de retorna valores numéricos? Vehiculo operator != no reutiliza el == Emprolijar
Vizgarra Livón, Daniel	SI	SI	A	SI	operator + no utiliza espacioDisponible
Wahler, Federico	SI	SI	A	SI	Vehiculo operator != no reutiliza el ==
Wybranski Romano, Cristian	SI	SI	D	SI	Emarca y Etamano pertenecen a Vehiculo. El programa no corre con el Main original. Vehiculo operator != no reutiliza el == Elimina un constructor de Sedan. Por qué? No cumple con el diagrama
Yaggi, Daniel	NO	NO	S/C	NO	Constructor Taller() debería ser privado
Ybarrola, Cecilia Susana	SI	SI	A	SI	Carpeta vacia
Zamudio, Carlos	SI	SI	D	SI	
Zanotti, Francisco	SI	SI	A	SI	No cumple lógica ni salida por pantalla Vehiculo operator != no reutiliza el == Constructores de Sedan no se reutilizan Tamano debía ser abstract Cambia orden del código para no castear en Mostrar()