

CLASSES. Composició (Revisió)

1.a.- Realitza una classe, de nom Examen, per a guardar informació sobre els exàmens d'un centre educatiu. La informació que es guarda d'un examen és: el nom de l'assignatura, l'aula, la data i l'hora. Per a guardar la data i l'hora cal realitzar dues classes, Data i Hora.

La classe Data guarda dia, mes i any. Tots els valors es reben en el constructor per paràmetre. A més, aquesta classe ha de tindre un mètode que retorne cadascun dels atributs i un mètode toString() que retorne la informació de la data en forma de String.

La classe Hora guarda hora i minut. També rep els valors per als atributs per paràmetre en el constructor, té mètodes que retornen cadascun dels atributs i un mètode toString().

1.b.- Realitza una aplicació que cree un objecte de tipus Examen, el mostre per pantalla, modifiqui la seua data i hora i ho torne a mostrar per pantalla.

2.a.- Escriu una classe Punt () que continga els següents mètodes:

a) els constructors i els mètodes getX(), getY(), setX(), setY() i distància()

b) Un mètode toString() que retorne la informació del Punt de la següent manera: (x,y).

2.b.- Utilitzant la classe Punt de l'exercici anterior, escriu una classe Polígon. Aquesta classe té com a atribut un array d'objectes Punt. Es consideren adjacents els objectes Punt que estiguen en cel·les consecutives del array i els punts que estan en la primera i última cel·la. Aquesta classe ha de tindre els següents mètodes:

a) El constructor, rebrà per paràmetre un array amb els objectes Punt que defineixen el Polígon.

b) traslladar(), rep per paràmetre el desplaçament en la coordenada x i el desplaçament en la coordenada y.

c) escalar(), rep per paràmetre el factor d'escala per a la coordenada x i el factor d'escala per a la coordenada y.

d) numVertex(), retorna el nombre de vèrtexs del Polígon.

e) toString(), retorna la informació dels punts del Polígon, un en cada línia.

f) perímetre(). Retorna el perímetre del polígon.

2.c.- Escriu una aplicació en la qual:

- a) Creu un Polígon de quatre vèrtexs, que seran (0,0), (2,0), (2,2) i (0,2) .
- b) Mostra la informació del polígon i el seu perímetre per pantalla.
- c) Traslladar el polígon 4 en l'eix x i -3 en l'eix y.
- d) Mostra la informació del polígon per pantalla.

HERÈNCIA

3.a.- Escriu una classe Multimèdia per a emmagatzemar informació d'objectes de tipus multimèdia (pel·lícules, discos, mp3...). Aquesta classe conté títol, autor, format i duració com a atributs. El format pot ser un dels següents: wav, mp3, midi, avi, mov, mp4, cdAudio i dvd. El valor de tots els atributs es passa per paràmetre en el moment de crear l'objecte. Aquesta classe té, a més, un mètode per a retornar cadascun dels atributs i un mètode toString() que retorna la informació de l'objecte. Finalment un mètode equals() que rep un objecte de tipus Multimèdia i retorna true en cas que el títol i l'autor siguin iguals als de l'objecte que anomena al mètode i false en cas contrari.

3.b.- Escriu una classe Pel·lícula que herete de la classe Multimèdia anterior. La classe Pel·lícula té, a més dels atributs heretats, un actor principal i una actriu principal. Es permet que un dels dos siga nul, però no els dos. La classe ha de tindre dos mètodes per a obtenir els dos nous atributs i ha de sobreescrivre el mètode toString() perquè retorne, a més de la informació heretada, la nova informació.

3.c.- Escriu una classe ListaMultimedia per a emmagatzemar objectes de tipus multimèdia. La classe ha de tindre un atribut que siga un array d'objectes Multimèdia i un enter per a comptar quants objectes hi ha emmagatzemats. A més, té un constructor i els següents mètodes:

a) el constructor rep per paràmetre un enter indicant el nombre màxim d'objectes que emmagatzemarà.

b) int size(): retorna el nombre d'objectes que hi ha en la llista.

c) boolean add(Multimèdia m): afeg l'objecte al final de la llista i retorna true, en cas que la llista estiga plena retornarà false.

d) Multimèdia get(int posició): retorna l'objecte situat en la posició especificada.

e) `int indexOf(Multimèdia m)`: retorna la posició de l'objecte que s'introdueix per paràmetre, si no es troba, retornarà -1.

f) `String toString()` retorna la informació dels objectes que estan en la llista.

3.d.- Escriu una aplicació on:

a) Es crea un objecte de tipus `ListaMultimedia` de grandària màxima 10.

b) Es demanen tres pel·lícules i s'afigen a la llista.

c) Es mostre la llista per pantalla.

d) Es creua un objecte de tipus `Pel·lícula` introduint el títol i l'autor d'una de les pel·lícules de la llista. Per a la resta dels arguments s'utilitzen valors no significatius.

e) Cerca la posició d'aquest objecte en la llista.

f) Obtinga l'objecte que està en aqueixa posició i el mostre per pantalla juntament amb la posició en la qual es troba.

3.f.- Escriu una classe `Disc` que herete de la classe `Multimèdia` ja realitzada. La classe `Disc` té, a part dels elements heretats, un atribut per a emmagatzemar el gènere al qual pertany (rock, pop, punk, etc.). La classe ha de tindre un mètode per a obtindre el nou atribut i ha de sobreescriure el mètode `toString()` perquè retorne, a més de la informació heretada, la nova informació.

3.g.- Escriu una aplicació on:

a) Es creua un objecte de tipus `ListaMultimedia` de grandària màxima 10.

b) Es creuen tres discos i s'afigen a la llista.

c) Es mostre la llista per pantalla.

d) Es creu un objecte de tipus `Disc` introduint el títol i l'autor d'un dels discos de la llista, per a la resta dels arguments s'utilitzen valors no significatius.

e) Cerca la posició d'aquest objecte en la llista.

f) Obtinga l'objecte que està en aqueixa posició i el mostre per pantalla juntament amb la posició en la qual es troba.

POLIMORFISME

4.- Escriu una aplicació on:

- a) Es creen un objecte de tipus ListaMultimedia de grandària màxima 10.
- b) Es creen tres discos i s'afigen a la llista.
- c) Es creen tres pel·lícules i s'afigen a la llista.
- d) Treballant amb la llista i suposant que no se sap en quines posicions estan els discos i les pel·lícules:
 - 1. Es mostre la llista per pantalla.
 - 2. Es calcule la duració total dels objectes de la ListaMultimedia.
 - 3. Es mostre quants discos hi ha de rock.
 - 4. S'obtinga quantes pel·lícules no tenen actriu principal.

ÀMBITS I VISIBILITAT

5.a.- Escriu una classe Cotxe de la qual heretaran CotxeCanviManual i CotxeCanviAutomatic. Els atributs dels cotxes són la matrícula, la velocitat i la marxa. Per a aquest exercici no es permet la marxa arrere, per tant no es permeten ni velocitat negativa, ni marxa negativa. En el constructor es rep el valor de la matrícula per paràmetre i s'inicialitza el valor de la velocitat i la marxa a 0. A més tindrà els següents mètodes:

getMatrícula(): que retorna el valor de la matrícula.

getMarxa(): retorna el valor de la marxa.

getVelocitat(): retorna el valor de la velocitat.

accelerar(): rep per paràmetre un valor en accelerar el cotxe.

frenar(): rep per paràmetre un valor en frenar el cotxe.

toString(): retorna en forma de String la informació del cotxe.

canviaMarxa(): rep per paràmetre la marxa a la qual s'ha de canviar el cotxe. Aquest mètode és protected, perquè puguin accedir a ell les classes que hereten de Cotxe, però no les classes d'altres paquets.

5.b.- La classe CotxeCanviManual sobreesciu el mètode canviaMarxa() i el fa públic, perquè pugui ser anomenat des de qualsevol classe. No permet que es canvie a una marxa negativa.

5.c.- La classe `CotxeCanviAutomatic` sobreescriu els mètodes `accelerar()` i `frenar()` perquè canvie automàticament de marxa conforme es va accelerant i frenant.

COMPATIBILITAT DE TIPUS

5.d.- Escriu una aplicació que demane per teclat la matrícula d'un cotxe i pregunte si el cotxe és amb canvi automàtic o no. Posteriorment, ha de crear un cotxe amb les característiques indicades per l'usuari i mostrar-lo. Accelerar el cotxe en 60 km/h, si és un cotxe amb canvi manual, canviar la marxa a tercera i tornar-ho a mostrar.

POLIMORFISME

6.a.- S'implementarà un simulador de Vehicles. Existeixen dos tipus de Vehicle: Cotxe i Camió.

a) Les seues característiques comunes són la matrícula i la velocitat. En el moment de crear-los, la matrícula es rep per paràmetre i la velocitat s'inicialitza a 0. El mètode `toString()` dels vehicles retorna informació sobre la matrícula i la velocitat. A més es poden accelerar, passant per paràmetre la quantitat en km/h que s'ha d'accelerar.

b) Els cotxes tenen a més un atribut per al nombre de portes, que es rep també per paràmetre en el moment de crear-lo. Té a més un mètode que retorna el nombre de portes.

c) Els camions tenen un atribut de tipus `Remolc` que inicialitza a null (per a indicar que no té remolc). A més té un mètode `posaRemolc()`, que rep el `Remolc` per paràmetre, i un altre `treuRemolc()`. Quan es mostre la informació d'un camió que porte remolc, a més de la matrícula i velocitat del camió, ha d'aparèixer la informació del remolc.

d) En aquesta classe cal sobreescriure el mètode `accelerar` de manera que si el camió té remolc i la velocitat més l'acceleració superen els 100 km/h es llance una excepció de tipus `MassaRapidException`.

e) Cal implementar la classe `Remolc`. Aquesta classe té un atribut de tipus sencer que és el pes i el valor del qual se li dona en el moment de crear l'objecte. Ha de tindre un mètode `toString()` que retorne la informació del remolc.

f) Implementar la classe `MassaRapidException`.

Implementar aquest disseny amb l'estructura més convincent.

6.b.- Utilitzant les classes de l'exercici anterior, implementa una aplicació que faci el següent:

- a) Declare i cree un array de 4 Vehicles.
- b) Cree 2 camions i 2 cotxes i els afija al array.
- c) Suposant que no se sap en quines cel·les els cotxes i en quins els camions:
 - 1. Pose un remolc de 5000 Kg als camions del array.
 - 2. Accelere tots els vehicles.
 - 3. Escriba per pantalla la informació de tots els vehicles.