

Programación II - 2011/2012 - Universitat Jaume I

Evaluación continua - Módulo 2 - Prácticas - Grupo B: LA5-LA6

26 de abril de 2012

La duración máxima de esta prueba es de 60 minutos. No puedes consultar libros ni apuntes.

Ejercicio 1 (3 puntos)

Supón que puedes utilizar una de tus implementaciones de la clase `Fecha` del módulo 2, que tiene los siguientes métodos públicos:

```
Fecha(int día, int mes, int año)
Fecha(Fecha otraFecha)
int getDía()
int getMes()
int getAño()
Fecha díaSiguiente()
Fecha sumarDías(int días)
String toString()
boolean equals(Object otroObjeto)
int compareTo(Fecha otraFecha)
static int díasMes(int mes, int año)
static boolean añoBisiesto(int año)
int restarFechas(Fecha otraFecha)
```

Escribe un método estático, externo a esa clase, que reciba como parámetro un vector de fechas y devuelva la más reciente, sabiendo que todas ellas son fechas pasadas. Si el vector está vacío debe devolver `null`.

Ejercicio 2 (3,5 puntos)

Considera que dispones de la clase `LíneaPoligonal` que has implementado en las prácticas del módulo 2 y de la clase `Punto` que has utilizado para ello. Recuerda que has utilizado un vector de puntos para representar internamente una línea poligonal. Además, ya has implementado los siguientes métodos:

```
LíneaPoligonal()
void añadir(Punto punto)
void quitar(int posición)
void quitar(Punto punto)
void mover(double desplazamientoX, double desplazamientoY)
double longitud()
String toString()
boolean equals(Object otroObjeto)
LíneaPoligonal crearCopiaEnProfundidad()
```

Incorpora a la clase `LíneaPoligonal` un nuevo método

```
double distancia(LíneaPoligonal otraLínea)
```

que devuelva la distancia entre las dos líneas poligonales, considerando que viene dada por la distancia entre los dos puntos más próximos de ambas. Puedes suponer que ambas líneas poligonales tienen al menos un punto.

Ejercicio 3 (3,5 puntos)

Incorpora a la clase `LíneaPoligonal` un nuevo método

```
LíneaPoligonal unir(LíneaPoligonal otraLínea)
```

de modo tal que al ejecutar `l3 = l1.unir(l2)` obtengamos en `l3` una nueva línea poligonal formada por los puntos de `l1` seguidos de los puntos de `l2`, sin que `l1` ni `l2` se vean modificadas. Además, cualquier transformación que se aplique más adelante a los puntos de `l3` no debe afectar a los de `l1` ni a los de `l2`.