

Pràctica 1: Implementant una classe

Programació Orientada a Objectes

Aquesta pràctica consisteix en acabar creant una matriu de distàncies entre diferents ciutats. El programa ha d'aconseguir emmagatzemar informació de les ciutats, concretament el seu nom i la localització i amb aquestes dades i calculant les distàncies pertinents, formar la matriu. Per fer-ho, hem creat quatre grans classes:

- **GeometricPoint:**

GeometricPoint ens serveix per crear punts geomètrics, els quals prendran el rol de ciutat, ja que aquesta al cap i a la fi també és un punt geomètric.

Com a atributs per aquesta classe tenim la posició tant de la coordenada X com de la Y i també un nom, que serà el de la ciutat. En quant als mètodes, primerament tenim el constructor, que utilitzarem per crear les instàncies amb l'assignació dels respectius valors, el conjunt de "setters" i "getters" i dues funcions: *distancePoint* i *printPoint*. La primera calcula la distància entre dos punts, mitjançant la norma entre aquests dos, i la segona mostra al terminal les coordenades d'un punt concret.

- **TestPoint:**

TestPoint l'hem utilitzat per comprovar que tots els mètodes de *GeometricPoint* funcionen segons l'esperat. Hem definit una serie de punts mitjançant el constructor els quals hem passat als diferents mètodes i els resultats els hem imprimit per pantalla.

```
Updating coordinates with setters:  
Position x before: 2.3  
Position x after: 34.0  
Position y before: 7.4  
Position y after: 4.9
```

```
Trying the PrintPoint method:  
Resulting point:(34.0, 4.9)
```

```
Trying the setName method:  
City name before: Barcelona  
City name after: Caceres
```

```
Trying the distancePoint method:  
Distance between p1 and p2: 30.46850833237492
```

- **DistanceMatrix:**

Aquesta classe és la responsable de preparar i crear la matriu de distàncies.

Com a atributs tenim una llista enllaçada ("LinkedList") de punts geomètrics (*GeometricPoint*) anomenada "cities", on es guarden les ciutats que formaran la matriu i una matriu tipus "double". En aquest cas el constructor només crea la "LinkedList" i com a mètodes tenim:

addCity, que afegeix un *GeometricPoint* prèviament creat, a la "LinkedList", *getCityName*, que donat un índex aplicat a la llista cities, retorna el nom de la ciutat, *getNoOfCities* retorna el tamany de la "LinkedList" mitjançant la funció "size()". Els altres dos mètodes ja fan referència a la creació de la matriu: *createDistanceMatrix* i *getDistance*. El primer crea i omple la matriu amb els valors de les distàncies entre les ciutats cridant a la funció definida

a la classe *GeometricPoint*, *distancePoint*. El de *getDistance* ens retorna un valor concret de la matriu especificat per dos índexs.

- **TestDistanceMatrix:**

A *TestDistanceMatrix* hem creat diferents punts geomètrics (ciutats) que hem anat afegint a la "*LinkedList*" i amb això hem pogut cridar a tots els mètodes de *DistanceMatrix* comprovant, com a *TestPoint*, tots els resultats per pantalla. En un d'aquests hem intentat simular la forma de la matriu de distàncies resultant, com es veu a la imatge.

```
Every city in LikedList:
Barcelona, Ankara, New York, Hong Kong,

Distance between Barcelona and Ankara: 6.685057965343306

Distance matrix:
0.0 6.685057965343306 86.80760335362335 33.46117152760794
6.685057965343306 0.0 80.47614553394068 27.073972741361768
86.80760335362335 80.47614553394068 0.0 61.680547987189605
33.46117152760794 27.073972741361768 61.680547987189605 0.0
```

Observacions i conclusió:

Durant la realització de la pràctica ens hem topat amb diversos problemes dins la classe *DistanceMatrix*.

Un dels problemes ha estat en la relació entre els atributs i el constructor: Com que els valors dels atributs en els dos casos no són d'un únic valor o instància i necessiten d'altres mètodes (que serien *addCity* i *createDistanceMatrix*) per ser omplerts, el constructor no pot tenir cap valor d'entrada i des del nostre punt de vista, això el fa prescindible.

A més, no hem sapigut fer funcionar a la vegada la definició de una "*LinkedList*" i matriu dins el constructor, ja que alhora d'utilitzar el mètode *getNoOfCities* per accedir al tamany de la llista enllaçada varem haver trobat un error que ens alerta que no podem accedir a aquest tamany perquè la definició del nostre constructor és "type double[][]". Per solucionar-ho hem hagut de treure la matriu del constructor i afegir-la al principi del mètode *createDistanceMatrix*. D'aquesta manera podem cridar a ".size()" per obtenir el tamany de la llista enllaçada "cities" sense que cap error.

Al veure els problemes que ens ha donat tot això hem pensat en una altre possible implementació de les classes de *DistanceMatrix* i *TestDistanceMatrix*. Podríem crear la "*LinkedList*" dins de *TestDistanceMatrix* en comptes de definir-la al constructor, deixant a aquest només amb la definició de matriu. Així podríem omplir la llista enllaçada amb el mètode *addCity* i els punts de les ciutats que hem creat, per posteriorment, passar-la per paràmetre a la funció *createDistanceMatrix*. Per tant dins de *createDistanceMatrix* no definiríem la matriu, ja que aquesta estaria definida al utilitzar el constructor.

A pesar de tot, finalment hem resolt correctament la pràctica, i els resultats obtinguts s'adeqüen amb el que es demanava a l'enunciat.