

Übung 10: Dynamische Datenstrukturen und Rekursion

Abgabetermin: 19. 01. 2010

Name: _____

Matrikelnummer: _____

Gruppe: G1 (Prähofer)

G2 (Wolfinger)

G3 (Wolfinger)

G4 (Jahn)

Aufgabe	Punkte	gelöst	abzugeben schriftlich	abzugeben elektro- nisch	Korr.	Pkte
Aufgabe 10.1	12	<input type="checkbox"/>	Java-Programm Testergebnisse	Java-Programm	<input type="checkbox"/>	
Aufgabe 10.2	12	<input type="checkbox"/>	Java-Programm Testergebnisse	Java-Programm	<input type="checkbox"/>	

Aufgabe 10.1: Sortierte lineare Liste

Implementieren Sie ein Verzeichnis der 100 reichsten Österreicher als sortierte lineare Liste. Lesen Sie die Namen und das Vermögen der Personen/Familien aus der Vorgabedatei *Top100.txt*. Die Datei ist nicht vor-sortiert und hat folgendes Format:

```
"Wlaschek, Karl & Familie"           3000000000
"Schmidt-Chiari, Guido & Familie"    2500000000
"Sohmen, Helmut"                     1300000000
"Breiteneder, Familie"               8500000000
"Doppler, Familie"                  3000000000
...
```

Erzeugen Sie für jede Person/Familie ein Objekt der Klasse *Person* und fügen Sie das Objekt nach dem Vermögen absteigend sortiert in die Liste ein. Implementieren Sie für die Liste eine Klasse *SortedList* mit folgender Schnittstelle:

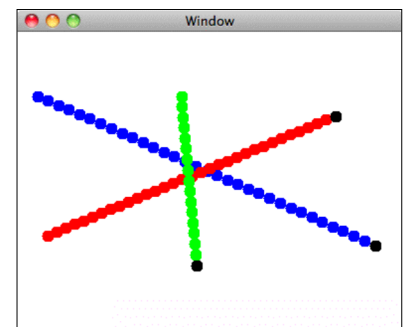
- *void insert(Person p)* fügt eine Person sortiert in die Liste ein
- *SortedList sublistAssetRange(long from, long to)* erzeugt eine neue Liste mit allen Personen deren Vermögen im Intervall *from-to* liegt
- *SortedList sublistNameRange(char from, char to)* erzeugt eine neue Liste mit allen Personen deren Name mit einem Buchstaben im Intervall *from-to* beginnt
- *String toString()* formatiert die sortierte Personenliste als Zeichenkette zur Ausgabe

Abzugeben sind: Java-Programm, Testergebnisse

Aufgabe 10.2: Rekursiver Linienalgorithmus

Implementieren Sie folgenden rekursiven Algorithmus der zwei Punkte $P_1(x_1, y_1)$ und $P_2(x_2, y_2)$ mit einer Linie verbindet in einer Methode *drawLine(Point p1, Point p2)*. Implementieren Sie die Klasse *Point*.

```
zeichneLinie(P1, P2)
  wenn P1 und P2 nahe beinander liegen
    zeichne die Punkte P1 und P2
  sonst
    berechne den Punkt M der die Gerade P1, P2 in der Mitte teilt
    zeichneLinie(P1, M)
    zeichneLinie(M, P2)
```



Die Klasse *Window.java* bekommen Sie auf der LVA-Homepage. Verwenden Sie nicht deren Methode *drawLine* - das Ziel dieser Aufgabe ist, dass Sie selbst eine Methode zum Zeichnen von Linien implementieren. Verwenden Sie zum Zeichnen der Punkte *drawPoint(int x, int y)*, oder *drawCircle(int x, int y, int r)*, wenn Sie die Punkte (siehe Abbildung) größer zeichnen möchten.

Abzugeben sind: Java-Programm, Testergebnisse