Esame di Programmazione II, 28 febbraio 2014

Esercizio 1 [**5 punti**] Si completi la seguente classe, che implementa un momento temporale preciso fino al minuto:

```
public class Time implements Comparable<Time> {
  ...
  public Time(int year, int month, int day, int hours, int minutes) {
    ...
  }

  @Override
  public int compareTo(Time other) {
    ...
  }

  public String toStringOnlyHour() {
    ... ritorna stringhe del tipo hh:mm (solo ore e minuti)
  }
}
```

Si noti che comparare due istanti temporali significa metterli in ordine temporale (chi viene prima nel tempo).

Esercizio 2 [**4 punti**] Si completi la seguente classe, che implementa un treno che parte a un certo istante temporale, ha un certo numero identificativo e una data destinazione:

```
public class Train implements Comparable<Train> {
  ...
  public Train(Time time, int number, String destination) { ... }

  @Override
  public String toString() {
    return String.format("%s : treno %d per %s", time.toStringOnlyHour(), number, destination);
  }

  @Override
  public int compareTo(Train other) {
    ... li mette in ordine temporale di partenza. A parita' di partenza, li mette
    in ordine crescente per numero di treno
  }
}
```

Esercizio 3 [**3 punti**] Si completi la seguente classe astratta, che rappresenta un cartellone che enumera (se iterato) i treni in partenza:

```
public abstract class Cartellone implements Iterable<Train> {

  @Override
  public final String toString() {
    ... ritorna la concatenazione del toString() di tutti i treni nel cartellone
  }
}
```

Esercizio 4 [**5 punti**] Si completi la seguente classe, che implementa un cartellone al quale è possibile aggiungere treni in partenza. I treni devono essere visualizzati in ordine crescente secondo il loro metodo `compareTo()`:

```
public class CartelloneModificabile extends Cartellone {
  ...
  public CartelloneModificabile() {}
  public void add(Train train) { ...aggiunge train a questo cartellone }
  public void add(Iterable<Train> trains) { ...aggiunge tutti i trains a questo cartellone }
```

```
  public void add(Train... trains) { ...aggiunge tutti i trains a questo cartellone }

  @Override
  public Iterator<Train> iterator() { ...ritorna un iteratore sui trains di questo cartellone }
}
```

Suggerimento: la classe di libreria `java.util.TreeSet` implementa un insieme che, se iterato, restituisce i suoi elementi in ordine crescente rispetto al loro `compareTo`. Ha un metodo `add(elemento)` che aggiunge l'elemento all'insieme.

Esercizio 5 [5 punti] Si completi la seguente classe, che implementa un cartellone che mostra solo i primi `max` treni (al massimo) mostrati da un altro cartellone. In altri termini, iterando su un `CartelloneLimitato` si ottengono i primi `max` treni (al massimo) del cartellone originale.

```
public class CartelloneLimitato extends Cartellone {
  ...
  public CartelloneLimitato(Cartellone original, int max) { ... }

  @Override
  public Iterator<Train> iterator() {
    ...ritorna un iteratore sui primi max treni di original (al massimo)
  }
}
```

Se tutto è corretto, l'esecuzione del seguente `main`:

```
public class Main {

  public static void main(String[] args) {
    CartelloneModificabile c = new CartelloneModificabile();

    // treno 1549 per Venezia delle 10:30 del 28/2/2014
    c.add(new Train(new Time(2014, 28, 2, 10, 30), 1549, "Venezia"));

    // treno 1804 per Bologna delle 11:45 del 28/2/2014
    c.add(new Train(new Time(2014, 28, 2, 11, 45), 1804, "Bologna"));

    // treno 1802 per Milano delle 11:45 del 28/2/2014
    c.add(new Train(new Time(2014, 28, 2, 11, 45), 1802, "Milano"));

    // treno 211 per Trieste delle 12:01 del 28/2/2014
    c.add(new Train(new Time(2014, 28, 2, 12, 01), 211, "Trieste"));

    // treno 1561 per Venezia delle 11:59 del 28/2/2014
    c.add(new Train(new Time(2014, 28, 2, 11, 59), 1561, "Venezia"));

    System.out.println(c);
    Cartellone lim = new CartelloneLimitato(c, 3);
    System.out.println("\nI prossimi 3 treni");
    System.out.println(lim);
  }
}
```

deve stampare:

```
10:30 : treno 1549 per Venezia
11:45 : treno 1802 per Milano
11:45 : treno 1804 per Bologna
11:59 : treno 1561 per Venezia
12:01 : treno 211 per Trieste


I prossimi 3 treni
10:30 : treno 1549 per Venezia
11:45 : treno 1802 per Milano
11:45 : treno 1804 per Bologna
```