

**Fondamenti di Informatica 1 - anno accademico 200? - 200?**  
**Prova di Programmazione, appello simulato**

Si vuole gestire un elenco di studenti realizzando un dizionario che contiene coppie di tipo “matricola nome”. Il campo “matricola” è un numero intero in formato long che ha la funzione di chiave per il dizionario (non possono esistere due matricole uguali), mentre il campo “nome” è una stringa composta da una o più parole che rappresenta il nome dello studente associato a ciascuna matricola.

Il candidato ha a disposizione il file **StudentiTester.java**, il quale contiene classi e interfacce da utilizzare e/o completare secondo quanto qui indicato:

- **Dictionary**. Interfaccia che rappresenta il tipo di dati astratto dizionario, contiene le firme dei metodi e le specifiche del loro funzionamento. *Non modificare*.
- **DictionaryItemNotFoundException**. Eccezione lanciata in caso di mancato ritrovamento di una coppia nel dizionario. *Non modificare*.
- **Studenti**. Classe che implementa l'interfaccia **Dictionary**. *Completare la classe* scrivendone la parte privata e realizzandone i metodi pubblici. La classe **Studenti** conterrà coppie di tipo “matricola nome” appartenenti alla classe **Pair**, realizzata come classe interna alla classe **Studenti** e il cui codice *non va modificato*. Si richiede che *il metodo find() abbia prestazioni  $O(\log n)$* . Si richiede inoltre di *realizzare un metodo toString per la classe Studenti*, che restituisca una stringa contenente gli elementi secondo il seguente formato: (1) ogni coppia “matricola nome” viene scritta su una riga diversa, e (2) all'interno di ogni riga la coppia viene scritta seguendo il formato specificato dal metodo toString della classe **Pair** (si veda il corpo di tale metodo).
- **StudentiTester**. Classe di collaudo. *Completare la classe* in modo che il suo metodo main realizzi il seguente comportamento:
  - Due nomi di file di testo, *file1* e *file2*, vengono passati come argomenti sulla riga di comando.
  - Vengono creati due oggetti di tipo **Studenti**. Nel primo si inseriscono elementi letti dal file *file1* (scritto nello stesso formato specificato sopra per il metodo toString di **Studenti**).
  - Viene richiesta l'immissione di un numero di matricola da input standard. Il numero immesso viene cercato nel primo elenco di studenti e la corrispondente coppia “matricola nome” viene (se trovata) spostata dal primo al secondo elenco di studenti (ovvero rimossa dal primo e inserita nel secondo elenco).
  - L'operazione descritta al punto precedente può essere ripetuta un numero non prefissato di volte. Il ciclo viene terminato tramite inserimento del carattere “Q” da input standard.
  - Al termine delle ricerche effettuate dall'utente, il contenuto del secondo elenco viene stampato sul file *file2* (nello stesso formato specificato sopra per il metodo toString di **Studenti**).

Nello svolgimento si possono usare, se ritenute utili, solo le seguenti classi della libreria standard:

- tutte le classi dei package java.lang e java.io;
- le classi Scanner, StringTokenizer e NoSuchElementException del package java.util.

Usando i file **iscritti.txt** e **input.txt** allegati, il programma può essere provato con il comando  
**java StudentiTester iscritti.txt promossi.txt < input.txt**

*Al termine della prova lasciare nella directory di lavoro tutti i file. La prima riga del file .java dovrà contenere un commento con nome/cognome del candidato, matricola, data, numero postazione. Prima di uscire dall'aula consegnare questa scheda compilata.*

Cognome e Nome:	Corso di Laurea:
Matricola:	Postazione: ADT
Consegno l'elaborato.	Non consegno l'elaborato e mi ritiro dall'esame.
<b>Firma</b>	<b>Firma</b>