

Esame Fondamenti di Informatica – 1° appello: 27 Gennaio 2022

La documentazione delle API di jdk può essere consultata. Se in dubbio, chiedete alla docente o ai sorveglianti se è ammesso l'utilizzo di una particolare classe e/o metodo.

Esercizio 1 prova di programmazione

Implementare una classe eseguibile il cui nome deve essere `CognomeMatricolaRic.java`, dove `Cognome` e `Matricola` devono essere il proprio nome e la propria matricola.

Importante: nella prima riga del file scrivere in un commento il proprio nome, cognome, numero di matricola.

Nel main si dovrà:

- chiedere all'utente di inserire da standard input una stringa `s` che potrà contenere lettere e numeri
- invocare un metodo statico ricorsivo (che dovrete implementare voi) che restituisca la concatenazione delle sole vocali presenti nella stringa, nell'ordine in cui appaiono in `s`
- invocare un metodo statico ricorsivo (che dovrete implementare voi) che restituisca la concatenazione delle sole vocali presenti nella stringa, in ordine inverso a quello in cui appaiono in `s`
- stampare in output i risultati mantenendo le vocali maiuscole/minuscole come apparivano originariamente in `s`

Ad esempio, con la stringa `s= "123superCaLIfragiliSTICHEspiralidoso456"` dovrete ottenere in uscita le stringhe:

Stringa solo con vocali: `uealaiiEiaioo`

Stringa solo con vocali, in ordine inverso: `ooiaiEliialaeu`

Esercizio 2 prova di programmazione

Sia data un'implementazione di un ADT mappa a dimensione fissata `FixedArrayMap` implementata come array riempito in parte.

Implementare:

- **una sua sottoclasse**, chiamata **PizzaMap**, con i metodi sotto definiti. La classe specializza la mappa in modo che contenga associazioni tra i nomi delle pizze e i rispettivi prezzi in euro. `PizzaMap` dovrà essere in grado di mantenere un numero di associazioni arbitrariamente grande.
- **una classe eseguibile `PizzeriaTesterCognomeMatricola` (dove “CognomeMatricola” va sostituito con i propri dati) che la utilizza** secondo le indicazioni date più sotto.

Oltre alla classe ***FixedArrayMap.java*** viene fornita la classe ***PizzaMapTester.class*** che permette di fare alcuni test sulla classe `PizzaMap`. I test non sono esaustivi: se tutto funziona non è detto che tutto sia giusto, ma se qualcosa non funziona, è probabile che ci sia un errore.

IMPORTANTE: nella prima riga di tutte le classi realizzate inserite anche un commento con cognome, nome, numero di matricola.

Metodi della classe `PizzaMap`, sottoclasse di `FixedArrayMap`:

```
/* Il metodo put verifica i pre-requisiti per una mappa che memorizza associazioni tra il
nome della pizza (una parola) e il prezzo associato (di tipo in grado di gestire numeri con
la virgola). Il metodo put di FixedArrayMap lancia l'eccezione FullMapException se la
mappa è piena. In questo caso l'eccezione deve essere catturata e gestita
ridimensionando opportunamente l'array p, prima di inserire la nuova associazione.
Il valore restituito è il valore precedentemente associato alla data chiave, se già
presente, oppure null, se la chiave è nuova nella mappa.
```

```
*/
```

```
public Object put(Object key, Object value)
```

```
/* Il metodo printMenu() stampa il contenuto della mappa, ordinando le pizze per prezzo.
L'array p, ovvero il contenitore dei dati della mappa, non deve essere modificato. Per
l'ordinamento potete utilizzare qualsiasi algoritmo visto durante il corso. La priorità è che
funzioni correttamente, in seconda battuta verrà valutata anche la complessità
dell'algoritmo utilizzato che dovrete specificare come commento nel codice.
```

```
*/
```

```
public String printMenu()
```

Classe eseguibile PizzeriaTesterCognomeMatricola

- Creare un oggetto PizzaMap
- Leggere dal file pizze.txt le associazioni e inserirle nella mappa. Il formato del file è: "nomePizza prezzo" su ciascuna riga. Si può assumere che il nome della pizza sia composto da una sola parola
- Stampare il contenuto della mappa
- Stampare il menù ordinato per prezzo
- Stampare il costo totale di una margherita, una bufala e due pizze al prosciutto
- Gestire tutte le possibile eccezioni
- Bonus per la lode: gestire nomi di pizze composti da più parole (per testare questa versione potete separare i nomi composti di alcune pizze nel file pizze.txt).

I prezzi devono preferibilmente essere stampati come verrebbero stampati in euro, ovvero con due cifre decimali.

Qui sotto potete vedere l'output del programma (con i nomi delle pizze considerati come unica parola). I punti e/o le virgole come separatori decimali dipendono dalle impostazioni del sistema. Se il file pizze.txt vi dà problemi con il separatore virgola ",", sostituitelo con il punto ".".

Contenuto della mappa

```
CrudoDiParma: 6.5
Prosciutto: 5.5
FruttiDiMare: 9.0
TrisDiFunghi: 8.5
Bufala: 6.0
Margherita: 4.0
QuattroFormaggi: 7.0
```

Menu' ordinato per prezzo

```
Margherita: 4.0
Prosciutto: 5.5
Bufala: 6.0
CrudoDiParma: 6.5
QuattroFormaggi: 7.0
TrisDiFunghi: 8.5
FruttiDiMare: 9.0
```

Prezzo per una margherita, una bufala e due pizze al prosciutto: 21,00