

Esame di Programmazione II, 2 settembre 2022

Si crei un progetto Eclipse e il package `it.univr.letters`. Si copino al suo interno le classi del compito. Non si modifichino le dichiarazioni dei metodi. Si possono definire altri campi, metodi, costruttori e classi, ma devono essere *private*. La soluzione che verrà consegnata dovrà compilare, altrimenti non verrà corretta.

L'interfaccia `Letters` è già completa e rappresenta una sequenza di lettere (cioè caratteri alfabetici):

```
public interface Letters {  
    int length(); // la lunghezza della sequenza  
    String toString(); // la concatenazione delle sue lettere  
    void forEach(Consumer<Character> command); // applica command alle sue lettere  
}
```

Si noti che il metodo `forEach` esegue del codice per ogni lettera della sequenza (dalla prima all'ultima della sequenza), e che questo codice è fornito a `forEach` tramite un parametro di tipo `java.util.function.Consumer`, un'interfaccia della libreria standard definita come segue:

```
public interface Consumer<T> {  
    void accept(T t); // fa qualcosa con t (cioe' lo "consuma")  
}
```

Esercizio 1 (12 punti) (si consegna `LowerCase.java`)

Si completi la classe `LowerCase.java`, un'implementazione di `Letters` che rappresenta una sequenza *minuscola*, cioè fatta da lettere minuscole dell'alfabeto inglese (sono ammesse ripetizioni).

Esercizio 2 (13 punti) (si consegna `Vulcanian.java`)

Si completi la classe `Vulcanian.java`, una sottoclasse di `LowerCase` che rappresenta una sequenza *vulcaniana*, cioè un caso particolare di sequenza minuscola che è fatta da due parti: la prima parte contiene vocali in ordine alfabetico; la seconda parte contiene consonanti in ordine alfabetico. Per esempio, una sequenza vulcaniana è `iobcchjjknnnnnprsstx`:

<u>io</u>	<u>bcchjjknnnnnprsstx</u>
vocali in ordine alfabetico	consonanti in ordine alfabetico

Invece `ioBcchjjknnnnnprsstx` non è vulcaniana perché non è neppure minuscola e per lo stesso motivo non lo è `io(cchjjknnnnnprsstx`. Neanche `ibocchjjknnnnnprsstx` è vulcaniana perché la vocale `o` segue la consonante `b`. Neanche `oibcchjjknnnnnprsstx` è vulcaniana perché le vocali non sono in ordine alfabetico. E infine neanche `iocbchjjknnnnnprsstx` è vulcaniana perché le consonanti non sono in ordine alfabetico.

Esercizio 3 (6 punti) (si consegna Main.java)

Si completi la classe Main.java.

Se tutto è corretto, l'esecuzione del Main.java stamperà qualcosa del tipo:

```
l1 = othaj
I suoi caratteri sono:
o
t
h
a
j

l2 = eeeuccfhjhjknqrvvwx
I suoi caratteri non italiani sono:
j
j
k
w
x
y

new LowerCase("tmwdmghbonqlwbwpiqkv") => successo
new LowerCase("tmwd(ghbonqlwbwpiqkv") => eccezione
new LowerCase("tmwDmghbonqlwbwpiqkv") => eccezione
new Vulcanian("iobcchjjknnnnnprsstx") => successo
new Vulcanian("ioBcchjjknnnnnprsstx") => eccezione
new Vulcanian("io(cchjjknnnnnprsstx") => eccezione
new Vulcanian("ibocchjjknnnnnprsstx") => eccezione
new Vulcanian("oibcchjjknnnnnprsstx") => eccezione
new Vulcanian("iocbchjjknnnnnprsstx") => eccezione
```