

## Fondamenti di Informatica Canale 1 - a.a. 2017-2018

### Prova pratica di programmazione del 25 Gennaio 2018

#### ORACOLO DI DELFI

Si desidera realizzare un programma che risponda alle domande dell'utente (come l'Oracolo di Delfi dell'antica Grecia). Le domande vengono acquisite da `stdin`, una riga alla volta, e per ciascuna viene stampata una risposta. Le domande devono essere formulate in modo che sia possibile rispondere affermativamente o negativamente. Il programma termina quando viene inserita una riga vuota.

Domande identiche devono restituire risposte identiche, come nell'esempio seguente:

```
$ Oracolo di Delfi, avro' una carriera professionale brillante?
$     Certamente si'.
$ Oracolo di Delfi, trovero' il grande amore?
$     Forse.
$ Oracolo di Delfi, prendero' 30 in questo esame?
$     Certamente si'.
```

L'ADT dizionario è definito dalla seguente interfaccia:

```
public interface Dictionary extends Container
{
    void insert(Comparable key, Object attribute);
    Object find(Comparable key);
    Object remove(Comparable key);
}
```

L'interfaccia `Container` è così definita:

```
public interface Container
{
    boolean isEmpty();
    void makeEmpty();
    int size();
}
```

Si scriva il codice della classe `D` che realizza l'interfaccia `Dictionary`:

```
public class D implements Dictionary {...}
```

Si scriva successivamente la seguente classe eseguibile `Oracolo` che eredita la classe `D`:

```
public class Oracolo extends D // classe Oracolo
{
    // parte privata
    ...
    // parte pubblica
    public static final String[] RISPOSTA = { /* v. file sorgente */ };
    public String rispondi(String domanda) {...}
    public static void main(String[] args) {...}
}
```

Il metodo `String rispondi(...)` elabora una risposta alla domanda passata come parametro esplicito al metodo. La strategia di elaborazione della risposta sia la seguente:

- il metodo effettua una ricerca nel dizionario con la chiave `String domanda`. Se la chiave è presente restituisce l'attributo della chiave; se la chiave non è presente genera casualmente una risposta tra quelle definite nella variabile `RISPOSTA` della classe `Oracolo` e inserisce la coppia `(domanda, risposta)` nel dizionario.

Per la generazione casuale si usi il metodo statico `random()` della classe `java.lang.Math`, il quale restituisce un valore `double` compreso nell'intervallo `[0,1)`. Quindi, come visto a lezione, per ottenere numeri interi pseudocasuali compresi nell'intervallo `[A,B]` si usi:

```
int n = (int)(A+(1+B-A)*Math.random());
```

Il metodo `main(...)` esegua le seguenti azioni:

- istanzia un esemplare della classe `Oracolo`;
- legge da `stdin` un insieme di domande, una per riga, finché non si trova una stringa vuota;
- a ciascuna domanda risponde invocando il metodo `rispondi(...)`
- stampa la domanda e la relativa risposta.

Il programma deve essere testato con il comando:

```
$ java Oracolo < domande.txt
```

dove il file `domande.txt` è allegato.

Gli elaborati prodotti dai candidati verranno provati con i seguenti comandi:

```
$ rm *.class
$ javac *.java
$ java Oracolo < domande.txt
```

Nell'esecuzione della prova si possono usare solo le classi dei pacchetti `java.lang` e `java.io` e le classi `Scanner` e `NoSuchElementException` del pacchetto `java.util`.

Non è lecito aggiungere metodi pubblici alle classi `D` e `Oracolo` oltre a quelli sopra definiti.

Alla fine della prova il candidato lascerà nella directory di lavoro i seguenti file: `Dictionary.java`, `Container.java`, `D.java`, `Oracolo.java`.

Sono valutati insufficienti i compiti che presentano errori di compilazione.

Prima di consegnare l'elaborato, completare l'intestazione della classe `Oracolo.java` con cognome e nome del candidato, numero di matricola, numero della postazione di lavoro, data.

---

### FONDAMENTI DI INFORMATICA Canale 1 - a.a. 2017-2018 - (prof. Sergio Canazza)

Nome e Cognome \_\_\_\_\_

Matricola \_\_\_\_\_ Postazione \_\_\_\_\_

☐ Consegno l'elaborato.      ☐ Non consegno l'elaborato e mi ritiro dall'esame.

Firma \_\_\_\_\_