

Descrizione delle classi e dei metodi da implementare

Di seguito è fornita una descrizione di dettaglio delle classi e dei metodi da implementare.

Classe AEROMOBILE

La classe Aeromobile definisce ed implementa l'entità omonima caratterizzata dalle proprietà *codice (String)* e *numeroSequenziale (int)*. Le suddette proprietà sono in sola lettura e possono essere solo impostate al momento in cui viene istanziato l'oggetto mediante il seguente costruttore:

```
public Aeromobile(String codice)
```

senza che vi sia più la possibilità di modificarle. Si noti che la proprietà *numeroSequenziale* viene calcolata direttamente dal costruttore in modo da garantire che non vi siano istanze di oggetti della classe *Aeromobile* (o di eventuali classe derivate) che abbiano lo stesso valore della proprietà suddetta.

La classe rende disponibili i metodi di accesso in lettura delle due suddette variabili di istanza. La classe sovrascrive il metodo *toString()* di *Object* in modo da restituire a titolo di esempio in relazione ad un aeromobile con numero sequenziale 5 e con codice DEIF671 la seguente stringa:

Aeromobile n. 5 - Codice = DEIF671

Non deve essere possibile istanziare direttamente oggetti della classe Aeromobile, ma solo oggetti di eventuali classi derivate.

Classi AEREOAMOTORE e ALIANTE

Le classi AereoAMotore e Aliante definiscono ed implementano le entità omonime estendendo la classe Aeromobile aggiungendo rispettivamente la proprietà *numRotori (int)* e *efficienza (double)*. La nuova proprietà in ogni classe come quelle ereditate dalla superclasse è in sola lettura, non modificabile dopo la creazione dell'oggetto che può avvenire tramite il costruttore:

```
public AereoAMotore(String codice, int numRotori)
```

```
public Aliante(String codice, double efficienza)
```

Entrambe le classi sovrascrivono il metodo *toString()* in modo da restituire a titolo di esempio la seguenti stringhe nel caso di aereo a motore:

Aeromobile n. 5 - Codice = DEIF671 - Tipo = Aereo a motore - Numero rotorì = 4 e nel caso di aliante:

Aeromobile n. 6 - Codice = IQ721 - Tipo = Aliante - Efficienza = 0.34

Interfaccia CODECHECKER

CodeChecker è un'interfaccia funzionale caratterizzata dal metodo:

```
public boolean check(String codice)
```

Le classi che implementano tale interfaccia s'impegnano a implementare il metodo restituendo *true* solo se una determinata condizione è verificata sulla stringa codice passata come parametro.

Classi CODECHECKIT e CODECHECKEN

Le due classi *CodeCheckIT* e *CodeCheckEN* implementano l'interfaccia *CodeChecker*.

Il metodo *check* della classe *CodeCheckIT* restituisce *true* se il codice è nel formato ITA^ADDD dove *A* indica un carattere alfabetico e *D* indica un carattere numerico; si riportano di seguito esempi di codici corretti per i quali il metodo restituirebbe *true*:

ITLm654 ITUU788 ITaR421 ITbv765

ed esempi di codici errati per i quali il metodo restituirebbe *false*:

itlF671 IT876 ITK871 ITJH7

Il metodo *check* della classe *CodeCheckEN* restituisce *true* se il codice è nel formato EN^{DD}A^A dove *A* indica un carattere alfabetico e *D* indica un carattere numerico; si riportano di seguito esempi di codici corretti per i quali il metodo restituirebbe *true*:

EN65Lm EN78UU EN23kj EN31aB

ed esempi di codici errati per i quali il metodo restituirebbe *false*:

enlF671 EN876 ENK871 EN7JH

Classe AEROPORTO

La classe *Aeroporto* definisce ed implementa un contenitore di oggetti della classe *Aeromobile* (e classi derivate). Per implementare tale comportamento incapsula un'opportuna collezione che consente agevolmente di realizzare le operazioni sotto riportate. La classe è caratterizzata dalla proprietà *checker* di tipo *CodeChecker* che viene inizializzata dal costruttore:

```
public Aeroporto(CodeChecker checker)
```

La classe rende disponibili i seguenti metodi:

```
public boolean inserisci(Aeromobile a)
```

- consente di inserire l'aeromobile passato come parametro; l'inserimento viene

effettuato solo nel caso in cui il codice soddisfi il criterio determinato dal *CodeChecker* impostato in fase di inizializzazione dell'aeroporto in tal caso restituendo *true*, altrimenti restituisce *false* senza effettuare l'inserimento. Nel caso non sia stato impostato alcun *CodeChecker* (*checker == null*) l'inserimento viene effettuato senza fare controlli e viene restituito *true*.

- *public Aeromobile cerca(String codice)*
 - consente di ricercare un aeromobile tramite il *codice*; in pratica restituisce *null* se non vi è alcun aeromobile con lo stesso codice, altrimenti restituisce una *shallow copy* dell'aeromobile trovato;
- *public Aeromobile rimuovi(String codice)*
 - consente di rimuovere un aeromobile tramite il *codice*; in pratica restituisce *null* se non vi è alcun aeromobile con lo stesso codice, altrimenti rimuove l'aeromobile trovato e ne restituisce il riferimento;
- *public int elementi()*
 - restituisce il numero di aeromobili presenti nell'aeroporto

La classe sovrascrive il metodo *toString()* di *Object* restituendo una stringa contenente tutti gli aeromobili presenti nell'istanza di *Aeroporto* come nell'esempio seguente:

Aeromobile n. 1 - Codice = KL234 - Tipo = Aereo a motore - Numero rotori = 4
Aeromobile n. 2 - Codice = ITMJ954 - Tipo = Aereo a motore - Numero rotori = 2
Aeromobile n. 3 - Codice = EN66IO - Tipo = Aereo a motore - Numero rotori = 1
Aeromobile n. 4 - Codice = ITPPM22 - Tipo = Aereo a motore - Numero rotori = 3

- *public void salvaDOS(String nomefile)*
 - consente di memorizzare in formato binario lo stato dell'oggetto *Aeroporto* codificando opportunamente gli attributi tramite un *DataOutputStream*
- *public static Aeroporto leggiDIS(String nomefile)*
 - consente di ricostruire lo stato dell'oggetto *Aeroporto* a partire dagli attributi - in maniera duale rispetto a quanto previsto nel metodo *salvaDOS(...)* - tramite un *DataInputStream*
- *public void salvaOBJ(String nomefile)*
 - consente di salvare l'oggetto *Aeroporto* in formato binario serializzato
- *public static Aeroporto leggiOBJ(String nomefile)*
 - consente di leggere l'oggetto *Aeroporto* salvato in formato binario serializzato

===== Test n. 1 - Aeroporto senza CodeChecker =====

Test inserimenti

Sono stati effettuati n. 10 inserimenti

Elementi inseriti:

Aeromobile n. 1 - Codice = KL234 - Tipo = Aereo a motore - Numero rotori = 4
Aeromobile n. 2 - Codice = ITmJ954 - Tipo = Aereo a motore - Numero rotori = 2
Aeromobile n. 3 - Codice = EN66IO - Tipo = Aereo a motore - Numero rotori = 1
Aeromobile n. 4 - Codice = ITPPM22 - Tipo = Aereo a motore - Numero rotori = 3
Aeromobile n. 5 - Codice = itIF671 - Tipo = Aereo a motore - Numero rotori = 4
Aeromobile n. 6 - Codice = IQ721 - Tipo = Aliante - Efficienza = 0.34
Aeromobile n. 7 - Codice = ITPO251 - Tipo = Aliante - Efficienza = 0.26
Aeromobile n. 8 - Codice = EN65IK - Tipo = Aliante - Efficienza = 0.44
Aeromobile n. 9 - Codice = ITOL966 - Tipo = Aliante - Efficienza = 0.16
Aeromobile n. 10 - Codice = EN19UIK - Tipo = Aliante - Efficienza = 0.16

Test ricerca

Ricerca in base al codice: KL234

Aeromobile trovato: Aeromobile n. 1 - Codice = KL234 - Tipo = Aereo a motore - Numero rotori = 4

Test cancellazione

Cancellazione codice: EN65IK

Aeromobile eliminato: Aeromobile n. 8 - Codice = EN65IK - Tipo = Aliante - Efficienza = 0.44

Numero elementi residui: 9

===== Test n. 2 - Aeroporto con CodeChecker IT =====

Test inserimenti

Sono stati effettuati n. 3 inserimenti

Elementi inseriti:

Aeromobile n. 2 - Codice = ITmJ954 - Tipo = Aereo a motore - Numero rotori = 2
Aeromobile n. 7 - Codice = ITPO251 - Tipo = Aliante - Efficienza = 0.26
Aeromobile n. 9 - Codice = ITOL966 - Tipo = Aliante - Efficienza = 0.16

Test ricerca

Ricerca in base al codice: KL234

Nessun aeromobile corrisponde al criterio di ricerca

Test cancellazione

Cancellazione codice: EN65IK

Nessun aeromobile corrisponde al criterio di ricerca

Numero elementi residui: 3

===== Test n. 3 - Aeroporto con CodeChecker EN =====

Test inserimenti

Sono stati effettuati n. 2 inserimenti

Elementi inseriti:

Aeromobile n. 3 - Codice = EN66IO - Tipo = Aereo a motore - Numero rotori = 1
Aeromobile n. 8 - Codice = EN65IK - Tipo = Aliante - Efficienza = 0.44

Test ricerca

Ricerca in base al codice: KL234

Nessun aeromobile corrisponde al criterio di ricerca

Test cancellazione

Cancellazione codice: EN65IK

Aeromobile eliminato: Aeromobile n. 8 - Codice = EN65IK - Tipo = Aliante - Efficienza = 0.44

Numero elementi residui: 1