

## SOLUZIONI

I candidati devono consegnare entro un'ora dall'inizio della prova.

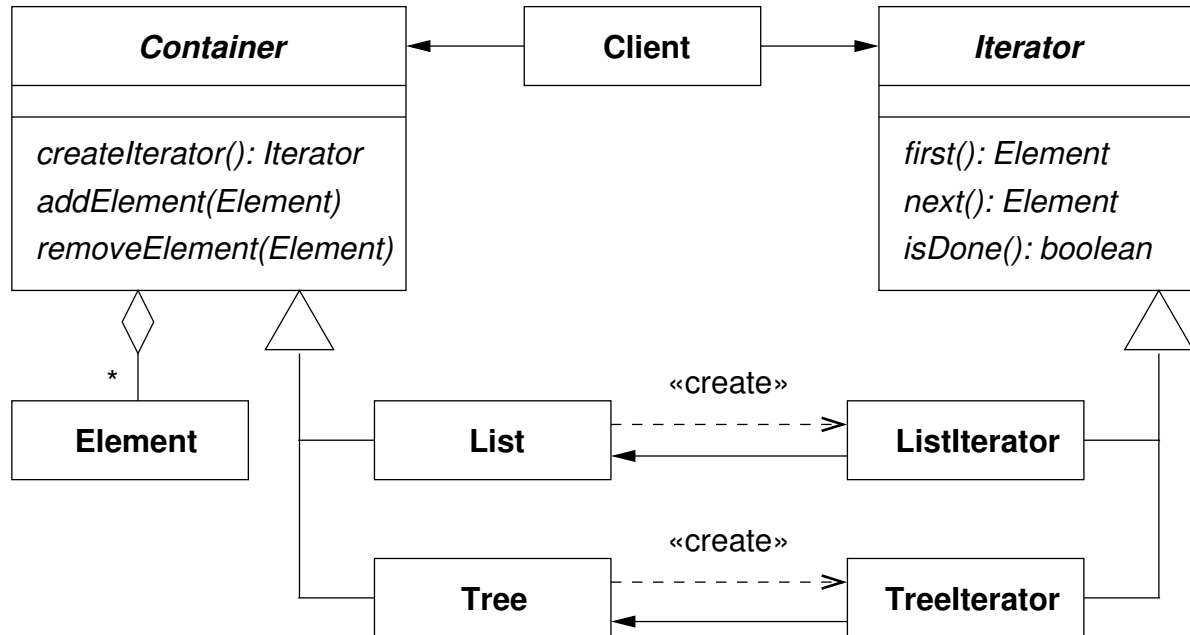


Fig. 1

**A1 In Fig. 1, ListIterator** ha operazioni che

- (a) restituiscono oggetti di tipo **List**. ☐
- (b) restituiscono oggetti di tipo **Element**. ☒
- (c) restituiscono oggetti di tipo **Container**. ☐

**A2 In Fig. 1, Client**

- (a) usa puntatori a **List**. ☐
- (b) usa puntatori a **ListIterator**. ☐
- (c) usa puntatori a **Container**. ☒

**A3 In Fig. 1, Tree**

- (a) implementa **Container**. ☒
- (b) implementa **Treeliterator**. ☐
- (c) usa **Container**. ☐

**A4 In Fig. 1, Client**

- (a) usa puntatori a **Treeliterator**. ☐
- (b) usa puntatori a **Listliterator**. ☐
- (c) usa puntatori a **literator**. ☒

**A5 In Fig. 1, createiterator()**

- (a) può essere implementata da **Container**. ☐
- (b) deve essere implementata da **List**. ☒
- (c) deve essere implementata da **literator**. ☐

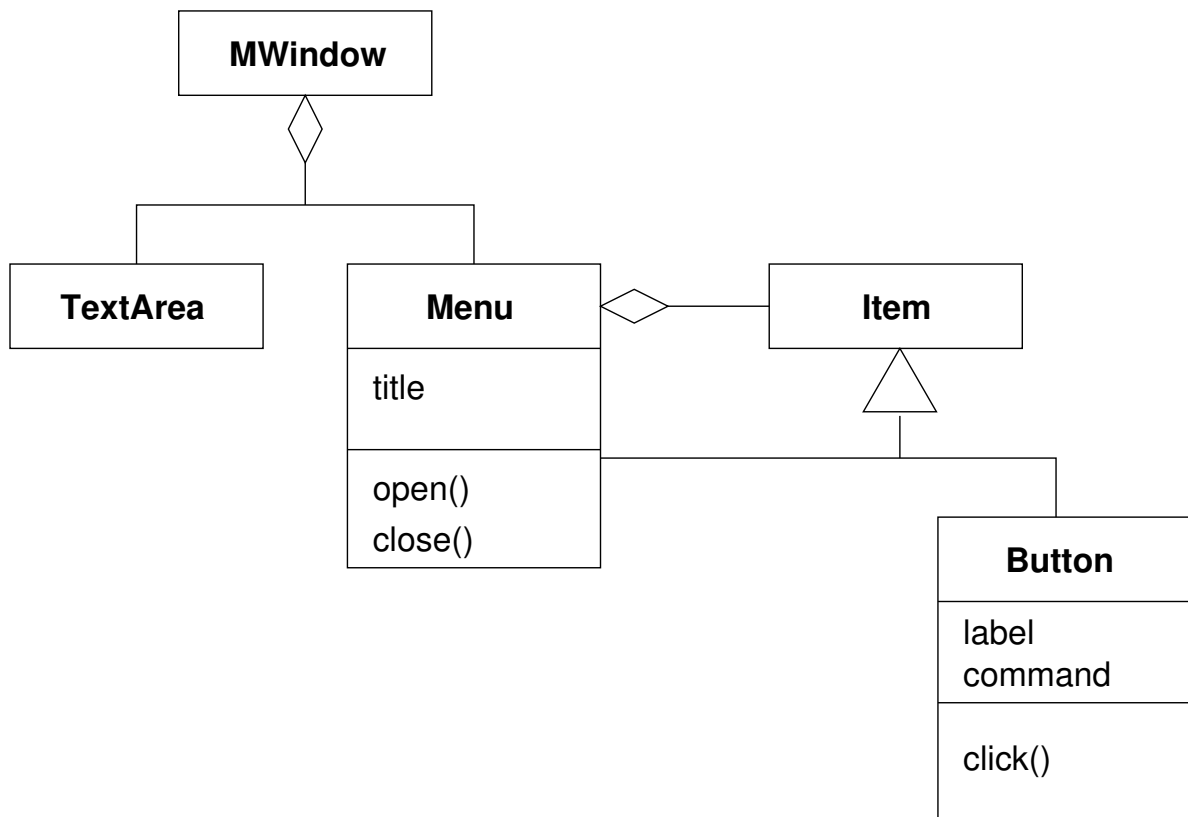


Fig. 2

**B1 In Fig. 2,**

- (a) Un oggetto **Menu** può contenere oggetti **Button** ☒
- (b) La classe **Menu** deriva dalla classe **Button** ☐
- (c) La classe **Menu** contiene la classe **Button** ☐

**B2 In Fig. 2,**

- (a) La classe **Menu** deriva dalla classe **Mwindow** ☐
- (b) Un oggetto **Mwindow** può contenere oggetti **Menu** ☒
- (c) Un oggetto **Menu** può contenere oggetti **Mwindow** ☐

**B3 In Fig. 2,**

- (a) Un oggetto **Button** può contenere oggetti **Menu** ☐
- (b) La classe **Button** deriva dalla classe **Item** ☒
- (c) La classe **Button** è base della classe **Item** ☐

**B4 In Fig. 2,**

- (a) La classe **Item** è base della classe **Button** ☒

- (b) La classe **Item** contiene la classe **Button** ☐
- (c) Un oggetto **Button** può contenere oggetti **Item** ☐

**B5 In Fig. 2,**

- (a) **Menu** eredita l'operazione **click** ☐
- (b) **Menu** eredita l'operazione **open** ☐
- (c) **Menu** implementa l'operazione **open** ☒

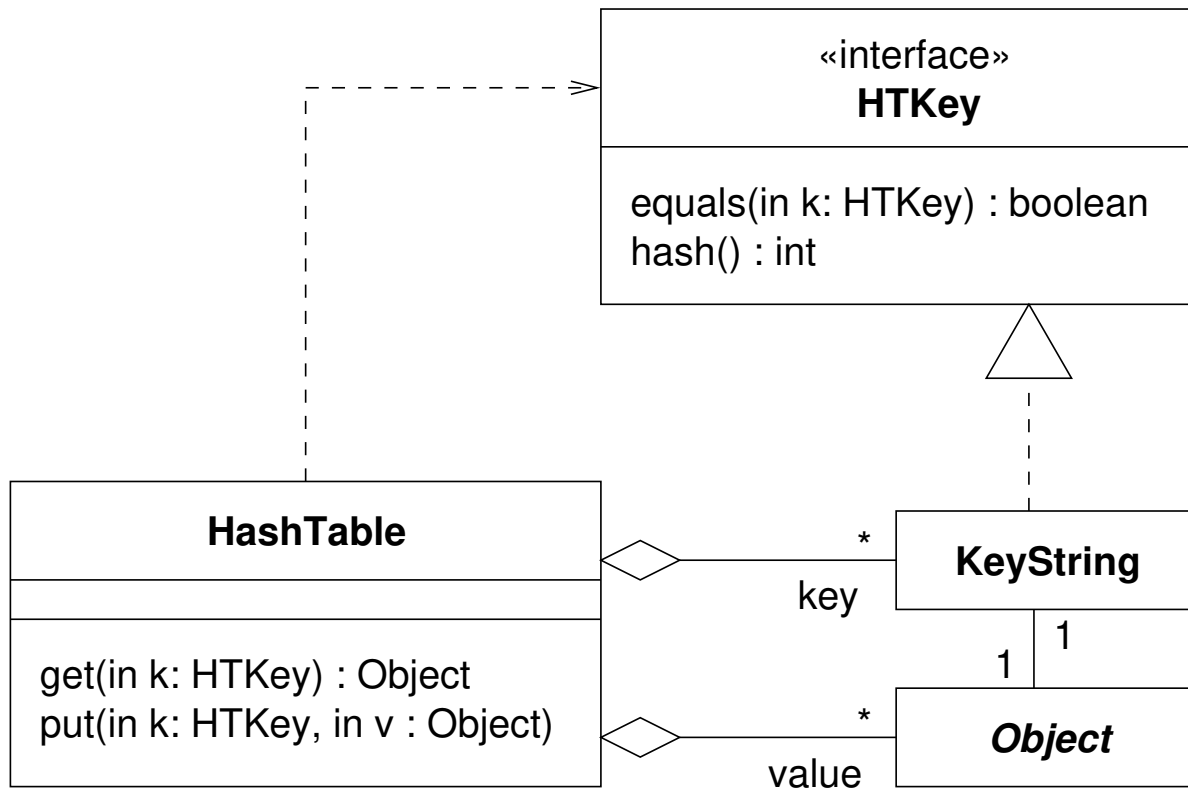


Fig. 3

**C1 In Fig. 3, HashTable**

- (a) implementa **HTKey**.
- (b) richiede **HTKey**.
- (c) offre **HTKey**.

☐  
☒  
☐

**C2 In Fig. 3, KeyString**

- (a) realizza **HTKey**.
- (b) dipende da **HTKey**.
- (c) appartiene a **HTKey**.

☒  
☐  
☐

**C3 In Fig. 3, lasciando HashTable immutata si può sostituire KeyString con un'altra classe?**

- (a) no, **HashTable** può usare solo chiavi **KeyString**.
- (b) sí, **HashTable** può usare chiavi di altro tipo.
- (c) sí, **HashTable** può usare chiavi di qualsiasi tipo.

☐  
☒  
☐

**C4 In Fig. 3, Object**

- (a) implementa **HashTable**.
- (b) deriva da **HashTable**.
- (c) appartiene a **HashTable**.

☐☐☒

**C5 In Fig. 3, put()**

- (a) è polimorfica.
- (b) è astratta.
- (c) è protetta.

☒☐☐

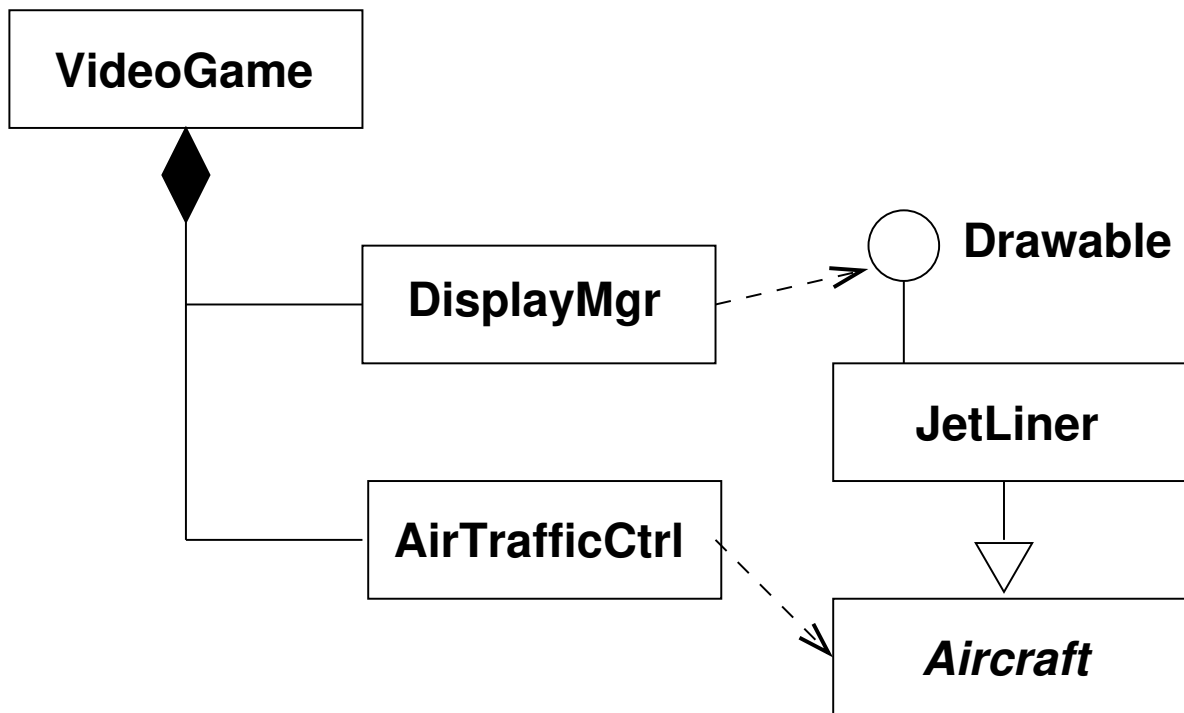


Fig. 4

D1 In Fig. 4,

- (a) **Drawable** è un'interfaccia richiesta da **Jetliner** ☐
- (b) **Drawable** è un'interfaccia implementata da **Jetliner** ☒
- (c) **Drawable** è un'interfaccia implementata da **Aircraft** ☐

D2 In Fig. 4,

- (a) **JetLiner** offre l'interfaccia di **Aircraft** ☒
- (b) **Aircraft** deriva da **JetLiner** ☐
- (c) **JetLiner** usa l'interfaccia di **Aircraft** ☐

D3 In Fig. 4,

- (a) **DisplayManager** offre l'interfaccia di **Drawable** ☐
- (b) **Drawable** è un'interfaccia richiesta da **DisplayMan-  
ager** ☒
- (c) **DisplayManager** deriva da **Drawable** ☐

D4 In Fig. 4,

- (a) **AirTrafficCtrl** fa parte di **VideoGame** ☒

- (b) **AirTrafficCtrl** fa parte di **DisplayManager** ☐
- (c) **VideoGame** fa parte di **AirTrafficCtrl** ☐

**D5** In Fig. 4,

- (a) **AirTrafficCtrl** implementa **Aircraft** ☐
- (b) **Aircraft** implementa **AirTrafficCtrl** ☐
- (c) **AirTrafficCtrl** usa **Aircraft** ☒



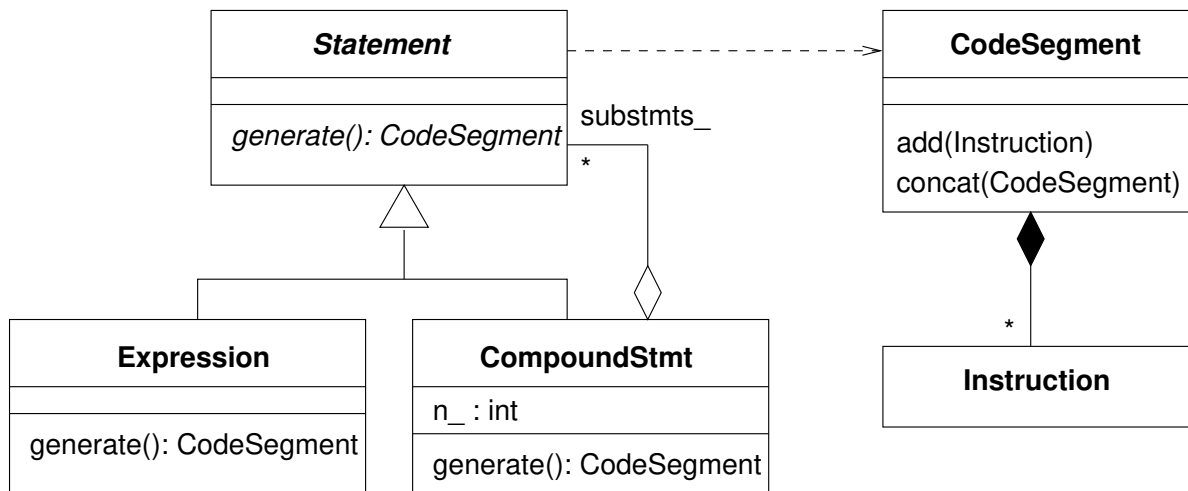


Fig. 5

E1 In Fig. 5,

- (a) *Statement* implementa **CodeSegment**. ☐
- (b) *Statement* dipende da **CodeSegment**. ☒
- (c) **CodeSegment** implementa *Statement*. ☐

E2 In Fig. 5,

- (a) una **Expression** può contenere dei **CodeSegment**. ☐
- (b) una **Expression** può contenere dei **CompoundStmt**. ☐
- (c) un **CompoundStmt** può contenere delle **Expression**. ☒

E3 In Fig. 5,

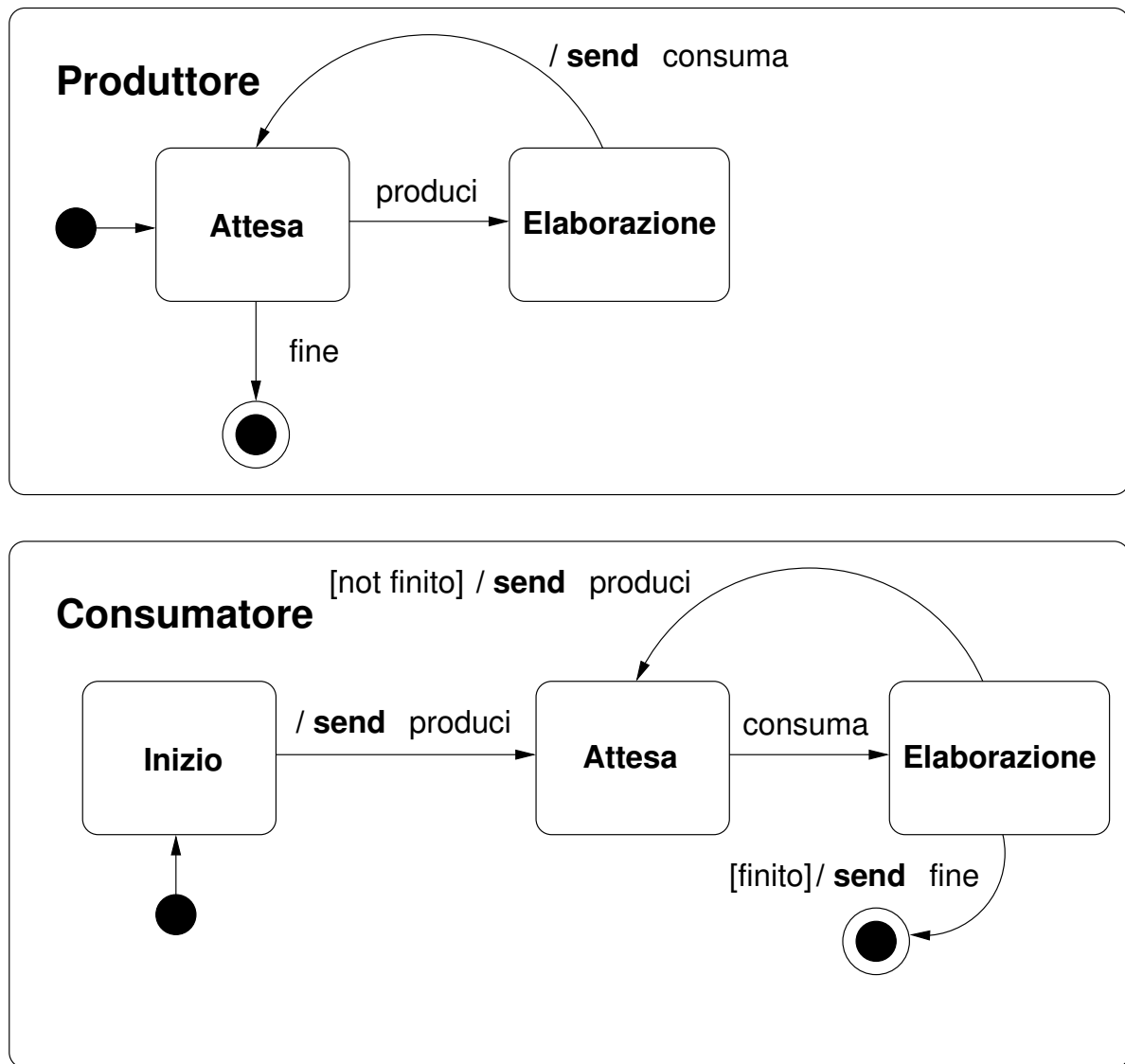
- (a) un **CompoundStmt** può contenere delle **Instruction**. ☐
- (b) un **CompoundStmt** può contenere dei **CodeSegment**. ☐
- (c) una **Instruction** fa parte di un **CodeSegment**. ☒

E4 In Fig. 5,

- (a) tutti gli *Statement* sono **Expression**. ☐
- (b) tutte le **Expression** sono *Statement*. ☒
- (c) tutti i **CompoundStmt** sono **Expression**. ☐

E5 In Fig. 5,

- (a) `generate()` restituisce un oggetto di tipo **CodeSegment**. ☒
- (b) `generate()` ha un argomento di tipo **CodeSegment**. ☐
- (c) `generate()` ha un argomento di tipo *Statement*. ☐



**Fig. 6**

Con riferimento alla Fig. 6, rispondere alle domande.

	<b>V</b>	<b>F</b>
<b>F1</b> il <b>Produttore</b> inizia l'interazione	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<b>F2</b> il <b>Produttore</b> fa terminare l'interazione	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<b>F3</b> il <b>Produttore</b> invia il segnale <b>consuma</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<b>F4</b> il <b>Consumatore</b> entra in <b>Elaborazione</b> prima del <b>Produttore</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<b>F5</b> il <b>Consumatore</b> entra in <b>Attesa</b> quando riceve <b>produci</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>