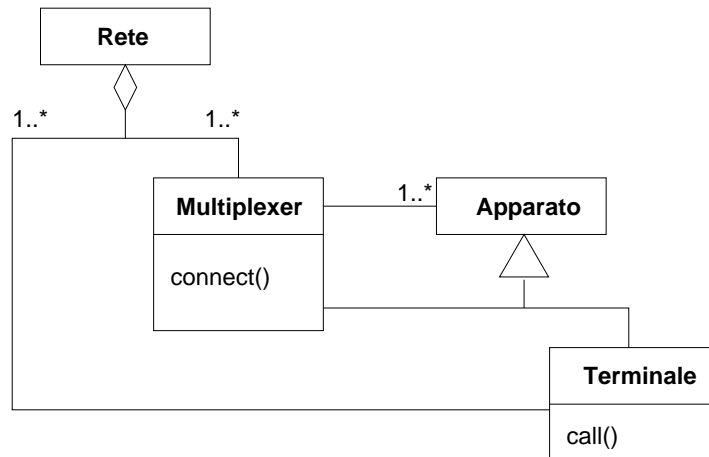


Scrivere le risposte (a, b, c oppure V, F) nelle rispettive caselle del file di testo allegato al messaggio inviato dal docente. I candidati devono consegnare entro 45 minuti dall'inizio della prova, inviando al docente il file di testo delle risposte, usando la funzione "rispondi" del cliente di posta elettronica. Chi si ritira dalla prova lo deve comunicare al docente per posta elettronica.



A1

- (a) Un'istanza di **Multiplexer** può essere collegato a istanze di **Terminale**. ☒
- (c) La classe **Multiplexer** contiene classe **Terminale**. ☐
- (b) La classe **Multiplexer** deriva dalla classe **Terminale**. ☐

A2

- (a) La classe **Multiplexer** deriva dalla classe **Rete**. ☐
- (c) Un oggetto **Multiplexer** può contenere oggetti **Rete**. ☐
- (b) Un oggetto **Rete** può contenere oggetti **Multiplexer**. ☒

A3

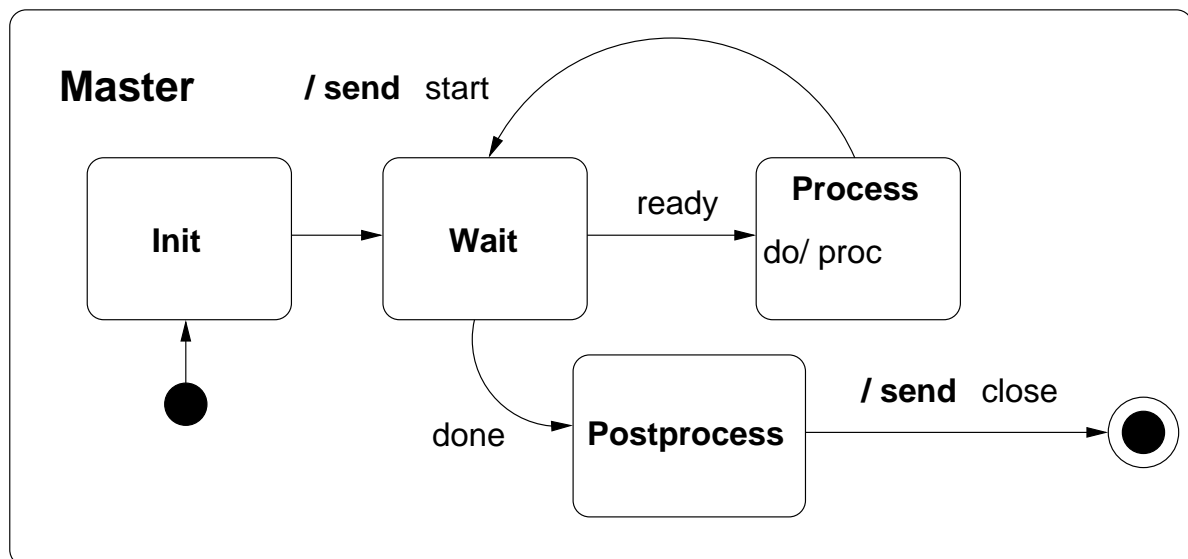
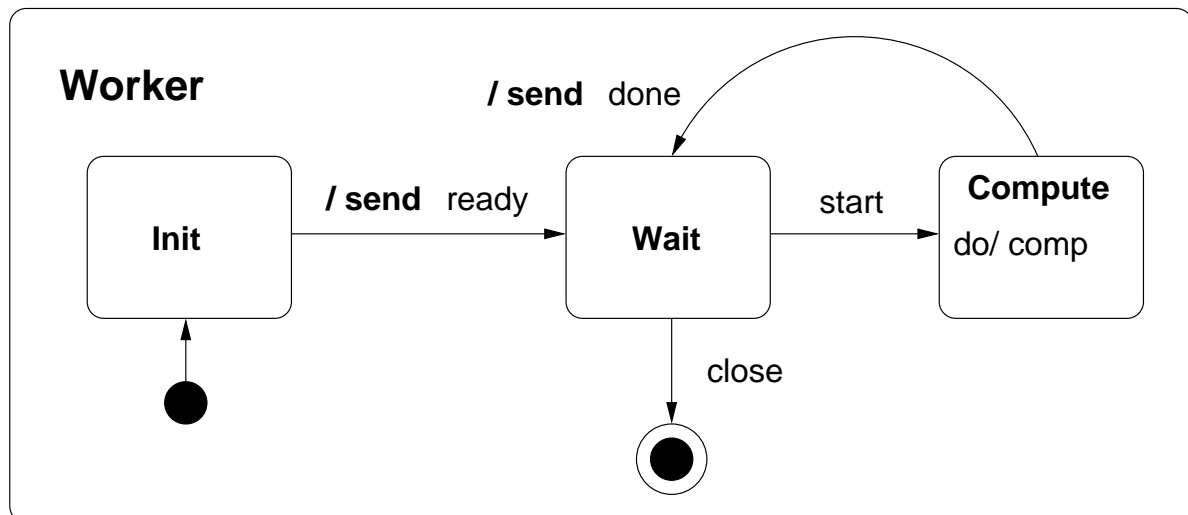
- (a) Un oggetto **Terminale** può essere collegato ad un oggetto **Multiplexer**. ☒
- (c) La classe **Terminale** è base della classe **Apparato**. ☐
- (b) La classe **Terminale** deriva dalla classe **Multiplexer**. ☐

A4

- (a) La classe **Apparato** è base della classe **Terminale**. ☒
- (c) Un oggetto **Terminale** può contenere oggetti **Apparato**. ☐
- (b) La classe **Apparato** contiene la classe **Terminale**. ☐

A5

- (a) **Multiplexer** eredita l'operazione **call**. ☐
- (c) **Multiplexer** offre l'operazione **connect**. ☒
- (b) **Multiplexer** eredita l'operazione **connect**. ☐



- | | V | F |
|--|-------------------------------------|-------------------------------------|
| B1 Il primo messaggio viene mandato da Worker . | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| B2 Il processo Master manda l'evento done . | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| B3 Il processo Worker può essere interrotto mentre esegue send done . | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| B4 L'attività comp segue l'attività proc . | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| B5 Il processo Master termina quando riceve l'evento done . | <input type="checkbox"/> | <input checked="" type="checkbox"/> |

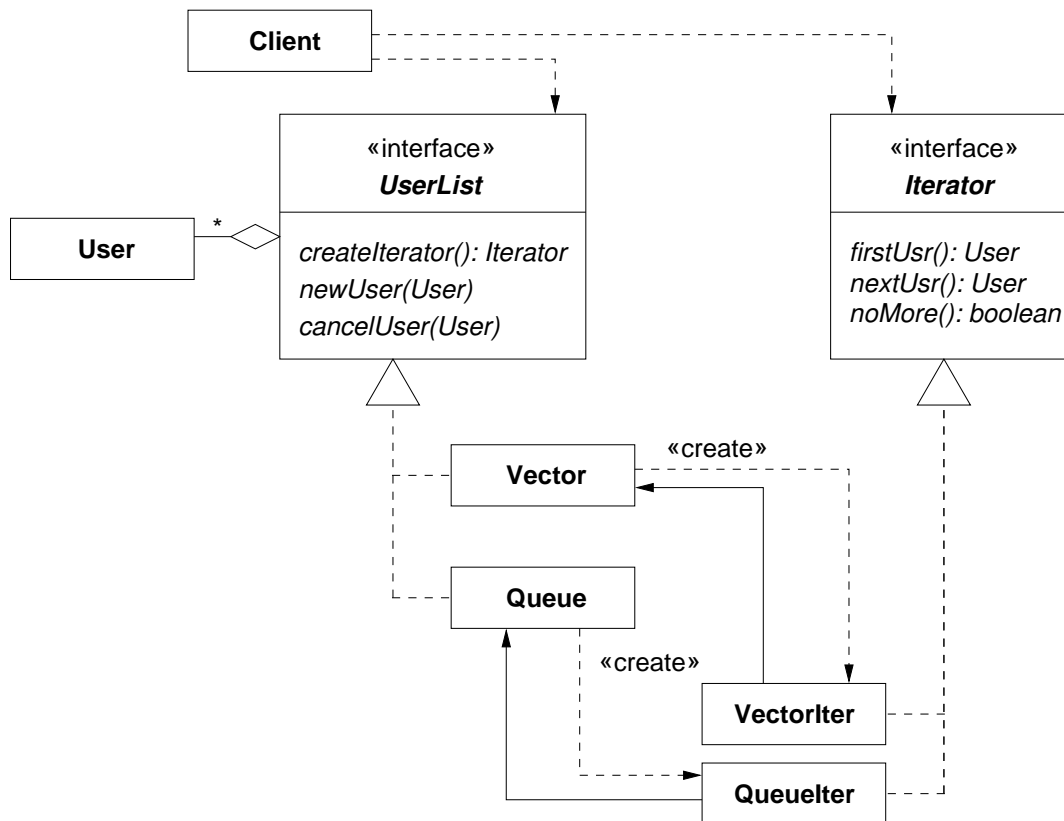
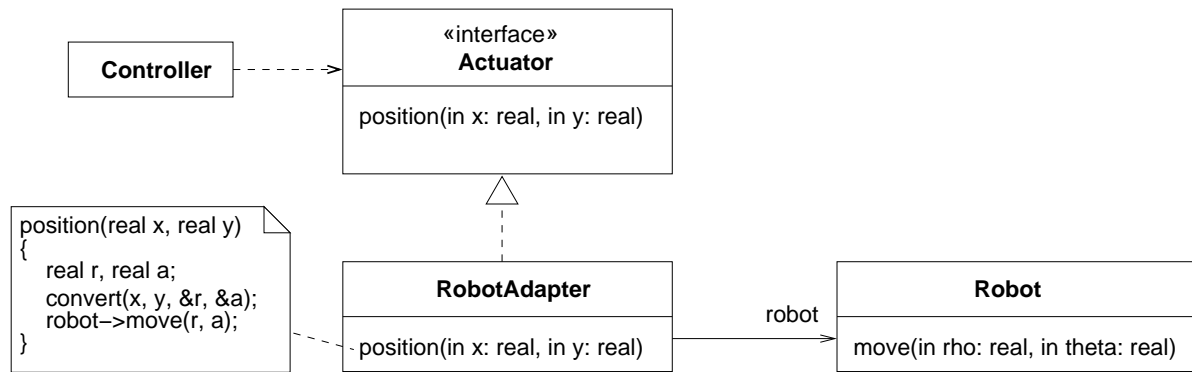
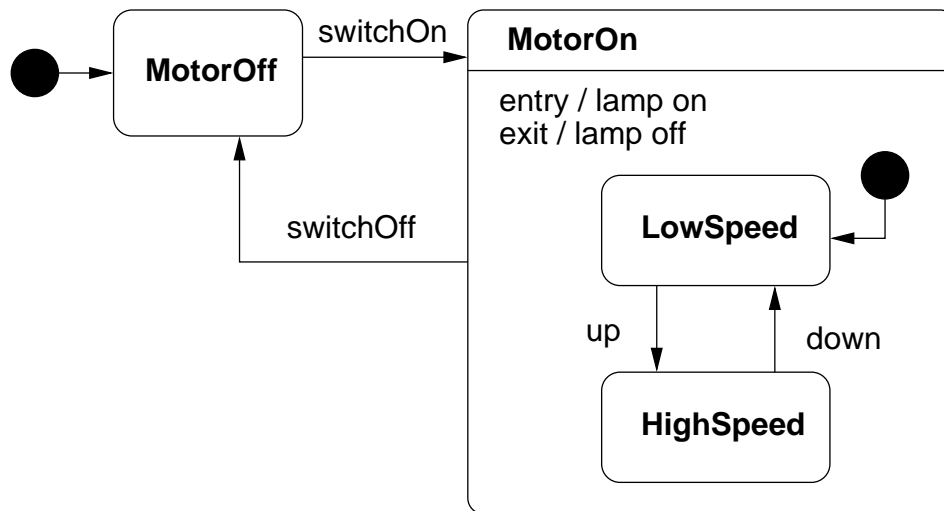


Fig. 4

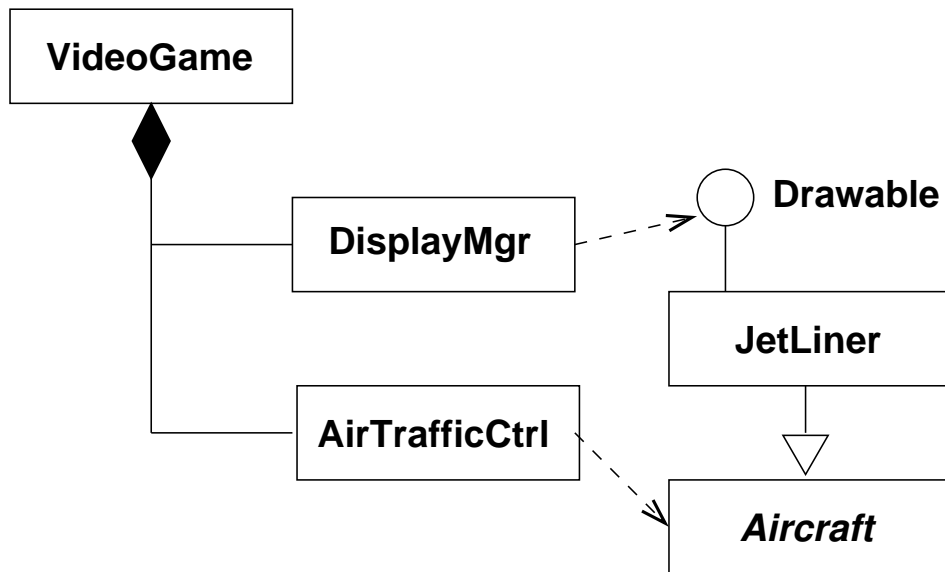
- | | | |
|-----------|---|-------------------------------------|
| C1 | VectorIter ha operazioni che | |
| | (a) restituiscono oggetti di tipo Vector . | <input type="checkbox"/> |
| | (b) restituiscono oggetti di tipo UserList | <input type="checkbox"/> |
| | (c) restituiscono oggetti di tipo User . | <input checked="" type="checkbox"/> |
| C2 | Client | |
| | (a) usa puntatori a Vector . | <input type="checkbox"/> |
| | (b) usa puntatori a UserList . | <input checked="" type="checkbox"/> |
| | (c) usa puntatori a VectorIter . | <input type="checkbox"/> |
| C3 | Queue | |
| | (a) realizza UserList . | <input checked="" type="checkbox"/> |
| | (b) usa UserList . | <input type="checkbox"/> |
| | (c) realizza QueueIter . | <input type="checkbox"/> |
| C4 | Client | |
| | (a) usa puntatori a QueueIter . | <input type="checkbox"/> |
| | (b) usa puntatori a Iterator . | <input checked="" type="checkbox"/> |
| | (c) usa puntatori a VectorIter . | <input type="checkbox"/> |
| C5 | createIterator() | |
| | (a) deve essere implementata da UserList . | <input type="checkbox"/> |
| | (b) deve essere implementata da Iterator . | <input type="checkbox"/> |
| | (c) deve essere implementata da Vector . | <input checked="" type="checkbox"/> |



- | | V | F |
|--|-------------------------------------|-------------------------------------|
| D1 Client usa l'interfaccia di Robot . | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| D2 RobotAdapter usa l'interfaccia di Robot . | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| D3 Actuator implementa Client . | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| D4 Robot realizza Actuator . | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| D5 Client dipende da Actuator . | <input checked="" type="checkbox"/> | <input type="checkbox"/> |



	V	F
E1 Lo stato iniziale del sistema è LowSpeed .	<input type="checkbox"/>	<input checked="" type="checkbox"/>
E2 L'evento switchOn causa sempre l'accensione del motore.	<input type="checkbox"/>	<input checked="" type="checkbox"/>
E3 Durante il funzionamento c'è sempre una luce accesa.	<input checked="" type="checkbox"/>	<input type="checkbox"/>
E4 Il motore si può spegnere (switchOff) solo nello stato LowSpeed .	<input type="checkbox"/>	<input checked="" type="checkbox"/>
E5 Il sistema non ha uno stato finale.	<input checked="" type="checkbox"/>	<input type="checkbox"/>



F1

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

F2

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

F3

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

F4

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

F5

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