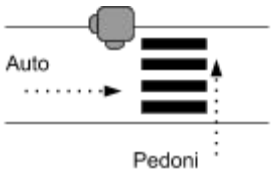
Programmazione 02/11/2015

Note sullo svolgimento della prova

- Non è consentito l'uso di vecchi testi d'esame, libri, o appunti.
- Utilizzare Notepad++ come editor e finestre DOS per la compilazione e esecuzione dei programmi.
- La cartella C:\prg contiene il JDK da utilizzare, la documentazione delle Java API e eventuali file ausiliari preparati dal docente.
- I file sorgenti relativi all'esercizio 1 devono essere posti nella cartella C:\prg\esercizio1 mentre quelli relativi all'esercizio 2 devono essere posti nella cartella C:\prg\esercizio2.
- Al termine della prova, dopo aver ritirato gli elaborati, il docente mostrerà una possibile soluzione. Dopo aver visto la soluzione, gli studenti avranno la possibilità di riprendere il proprio elaborato.

Esercizio 1



Una *Strada* è percorsa da auto e attraversata da pedoni. La strada è dotata di un attraversamento pedonale intelligente regolato da un semaforo a tre colori (verde, giallo, rosso). Quando il semaforo è verde o giallo per le auto, è rosso per i pedoni (e viceversa). La strada può essere percorsa da una sola auto alla volta, quando non è occupata da qualcun altro e il semaforo per le auto è verde. Viceversa, quando un'auto arriva al semaforo e il colore per le auto è giallo o rosso, l'auto si ferma in attesa del verde. Lo stesso vale per i pedoni.
Normalmente il semaforo è verde per le auto e rosso per i pedoni. Quando però ci sono pedoni in attesa di attraversare la strada, si innesca una sequenza di operazioni volta a permettere loro l'attraversamento: il semaforo diventa immediatamente giallo per le auto e rimane giallo per 5 secondi; quindi diventa rosso per le auto per 5 secondi (e di conseguenza verde per i pedoni). Trascorso questo tempo il semaforo torna nel suo stato iniziale (verde per le auto, rosso per i pedoni).
Realizzare una classe *Strada* che opera secondo le specifiche di cui sopra e dotata almeno dei seguenti metodi:

- *void arrivaAuto()* un'auto deve percorrere la strada.
- *void esceAuto()* un'auto ha percorso la strada..
- *void arrivaPedone()* un pedone deve attraversare la strada.
- *void escePedone()* un pedone ha attraversato la strada.

Sono presenti delle implementazioni di base delle classi *Auto*, *Pedone,* e *Prova* (che è possibile modificare a piacere). I file relativi sono nella cartella *esercizio1*.

Esercizio 2

Un sistema di *skill search* consente la ricerca di competenze specifiche, attraverso la consultazione di agenzie del lavoro, secondo un dato protocollo. Si consideri un numero di *agenzie del lavoro* non noto a priori, in comunicazione attraverso una configurazione ad anello. Il protocollo inizia con una *RichiestaDiCompetenze*, creata e inviata da una *AgenziaDelLavoro* (richiedente) alla successiva (ricevente). La *RichiestaDiCompetenze* è composta dalla descrizione delle competenze richieste, dalle porte di ascolto delle agenzie richiedente e proponente (quest'ultima stabilita nel primo giro di richiesta). Ogni agenzia ricevente, dopo aver visualizzato la richiesta, la invierà all'agenzia successiva se non ha una proposta, e cosi via. Non appena una agenzia ricevente ha una proposta, inserisce la propria porta di ascolto nell'oggetto *RichiestaDiCompetenze* e invia il medesimo all'agenzia richiedente invece che all'agenzia successiva. Quando l'agenzia richiedente riceverà la richiesta, la visualizzerà e la rimanderà all'agenzia ad essa successiva, e cosi via per il secondo giro. Nel secondo giro ogni agenzia riceve l'oggetto, lo visualizza, lo inoltra all'agenzia successiva e termina. Il secondo giro si ferma quando l'agenzia richiedente riceve nuovamente l'oggetto.

Si realizzi un'applicazione Java distribuita composta dalle classi *RichiestaDiCompetenze* e *AgenziaDelLavoro*. La classe *RichiestaDiCompetenze* mantiene ed elabora i dati suddetti, e raccoglie le offerte. La classe *AgenziaDelLavoro* svolge le operazioni di trasmissione di istanze di *RichiestaDiCompetenze* tra una agenzia e la successiva, in accordo al protocollo di cui sopra.

Attenersi esattamente a quanto espresso dalle seguenti figure. Fig.1 mostra una configurazione di test con tre agenzie. Fig.2 mostra i file dell'applicazione. Fig.3 presenta le classi da realizzare (in grigio), con i relativi campi e i metodi da creare, i package e le classi da importare e usare. Gestire la chiusura di flussi e connessioni aperti. Catturare solo le eccezioni obbligatorie e gestirle semplicemente stampando le relative informazioni. Fig.4 mostra il file di testing *make.bat*, con tre casi di test. Si noti che il richiedente si distingue per avere un "?" e il proponente per avere un "!". Fig.5 mostra la sequenza di passi (ad alto livello) che le istanze di *AgenziaDelLavoro* svolgono in un caso di test. La logica di *AgenziaDelLavoro* e *RichiestaDiCompetenze* dovrà essere valida a prescindere dai numeri di porta e dal numero di agenzie.

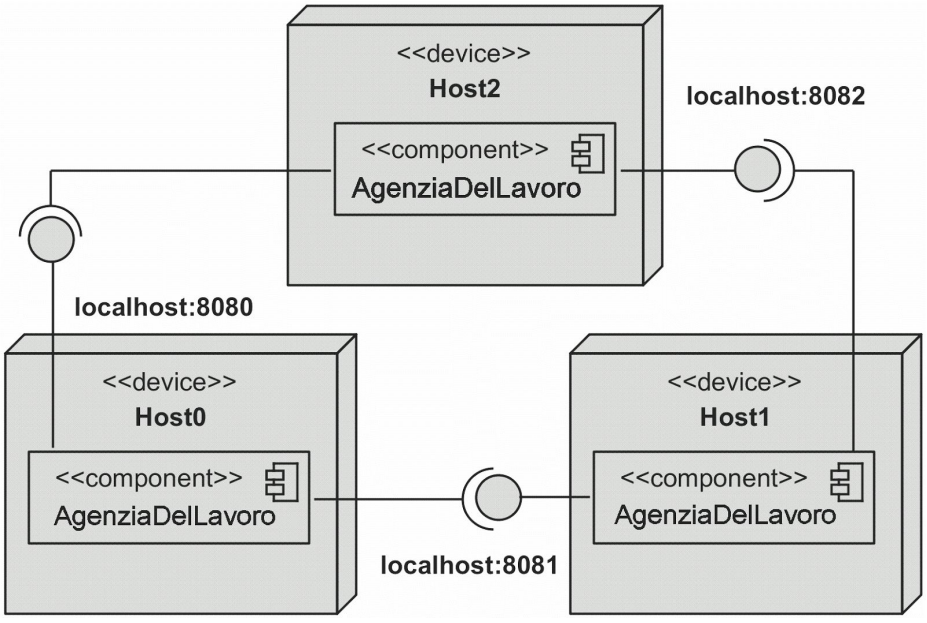
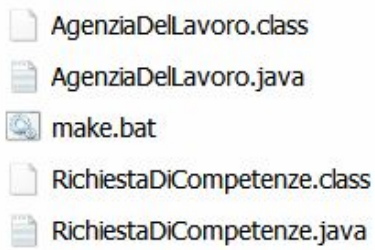


Fig.1 - Tre istanze di *AgenziaDelLavoro* e relative porte di ascolto



(viene fornito solo make.bat)

Fig.2 - c:\prg\esercizio2

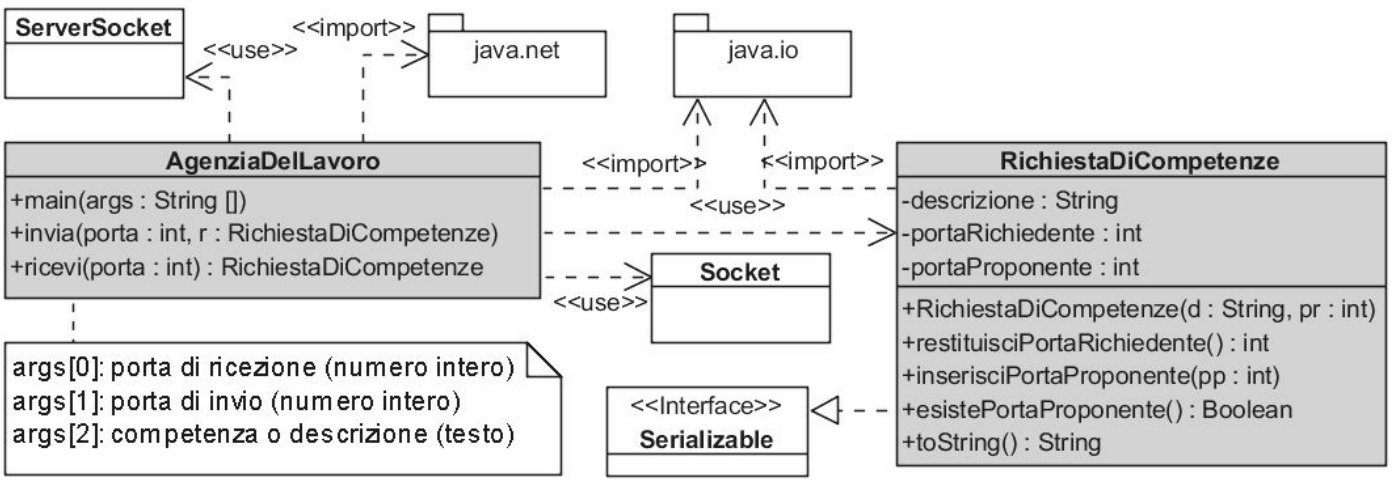


Fig.3 - Classi di cui deve essere composta l'applicazione (in grigio) e classi/package da usare (in bianco)

```
@echo off
c:\prg\jdk8\bin\javac *.java
pause

start cmd /k "color 2F && c:\prg\jdk8\bin\java AgenziaDelLavoro 8082 8080"
pause
start cmd /k "color 2F && c:\prg\jdk8\bin\java AgenziaDelLavoro 8081 8082"
pause
start cmd /k "color 2F && c:\prg\jdk8\bin\java AgenziaDelLavoro 8080 8081 "Programmatore C?""
pause

start cmd /k "color 3F && c:\prg\jdk8\bin\java AgenziaDelLavoro 8082 8080"
pause
start cmd /k "color 3F && c:\prg\jdk8\bin\java AgenziaDelLavoro 8081 8082 "Programmatore C!""
pause
start cmd /k "color 3F && c:\prg\jdk8\bin\java AgenziaDelLavoro 8080 8081 "Programmatore C?""
pause

start cmd /k "color 4F && c:\prg\jdk8\bin\java AgenziaDelLavoro 8082 8080 "Programmatore C!""
pause
start cmd /k "color 4F && c:\prg\jdk8\bin\java AgenziaDelLavoro 8081 8082"
pause
start cmd /k "color 4F && c:\prg\jdk8\bin\java AgenziaDelLavoro 8080 8081 "Programmatore C?""
pause
```

Fig.4 - File make.bat, da non modificare.

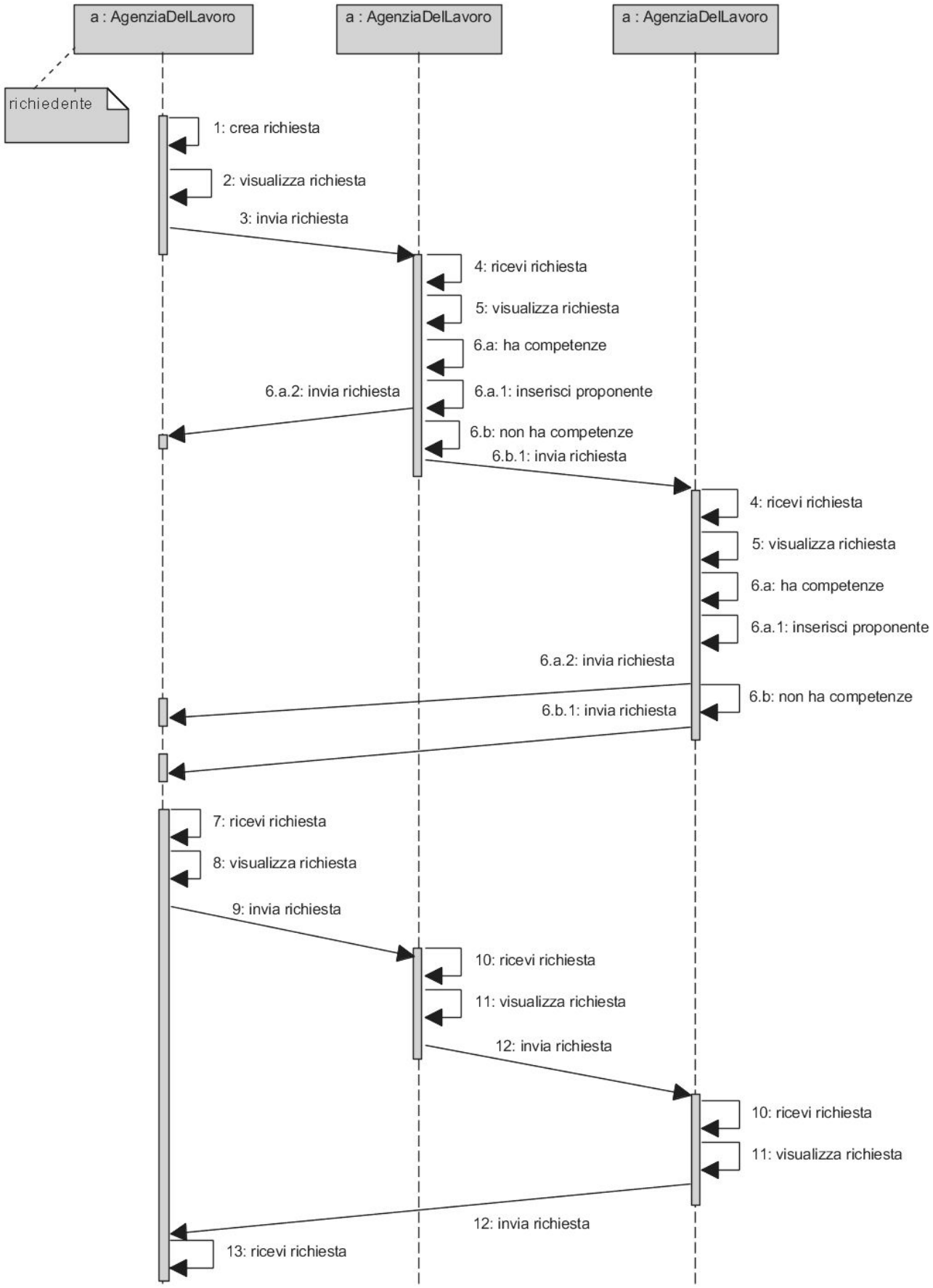


Fig.5 - Sequenza di passi che le istanze di AgenziaDelLavoro devono svolgere in un caso di test. Per brevità non viene rappresentata la classe RichiestaDiCompetenze.