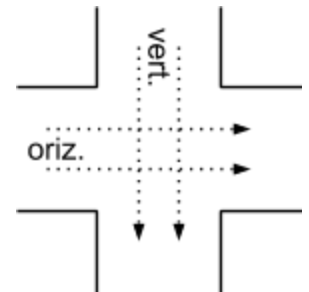


Note sullo svolgimento della prova

- 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

Un *Incrocio* è composto da due strade, una verticale e l'altra orizzontale. L'incrocio è dotato di un semaforo a tre colori: verde, giallo, rosso. Quando il semaforo è verde o giallo in una direzione, è rosso nell'altra direzione. Il semaforo cambia colore a tempo, indipendentemente dalla presenza di veicoli. Per semplicità il tempo di visualizzazione dei vari colori è sempre lo stesso ed è uguale a dieci secondi. In pratica il semaforo opera come indicato dal seguente esempio: per dieci secondi il semaforo è verde in direzione orizzontale e rosso in direzione verticale; seguono dieci secondi in cui il semaforo è giallo in direzione orizzontale e rosso in direzione verticale; seguono dieci secondi in cui il semaforo è rosso in direzione orizzontale e verde in direzione verticale; nei dieci secondi successivi il semaforo è rosso in direzione orizzontale e giallo in direzione verticale; ricomincia da capo.



L'incrocio può essere attraversato da un solo veicolo alla volta, quando l'incrocio non è occupato da un altro veicolo e il semaforo lungo la direzione del veicolo è verde. Viceversa, quando un veicolo arriva al semaforo e il colore lungo la sua direzione è giallo o rosso, il veicolo si ferma in attesa del verde.

I veicoli sono di due tipi: auto e bus. Le auto si comportano come sopra indicato. I bus hanno il potere di alterare i tempi del semaforo: quando un bus si ferma lungo una direzione, se il semaforo è verde lungo l'altra direzione e non ci sono bus lungo l'altra direzione, allora il verde lungo l'altra direzione diventa immediatamente giallo.

Realizzare una classe *Incrocio* che opera secondo le specifiche di cui sopra e dotata almeno dei seguenti metodi:

- *void arrivo(Direzione d, TipoVeicolo tv)* un veicolo arriva all'incrocio. Gli argomenti *d* e *tv* indicano rispettivamente la direzione e il tipo del veicolo. Il metodo può essere bloccante.
- *void esco()* un veicolo esce dall'incrocio.

I tipi *Colore*, *Direzione*, e *TipoVeicolo* sono già definiti. Sono presenti anche delle implementazioni di base delle classi *Veicolo*, *Auto* e *Bus* (che è possibile modificare a piacere). I file relativi sono nella cartella *esercizio1*.

Esercizio 2

Un sistema di votazione elettronica consente la votazione di proposte tramite la rete Internet. Si consideri un numero di *Votanti* non noto a priori, in comunicazione secondo una configurazione ad anello. Una votazione inizia con una *proposta* (il cui contenuto è rappresentato con del testo breve) votata e inviata da un votante al successivo, che a sua volta voterà e invierà la proposta al successivo, e così via. Quando il proponente riceverà la proposta, la elaborerà e visualizzerà il risultato, il quale verrà poi visualizzato e inviato da un votante al successivo, fino a giungere nuovamente al proponente.

La proposta può essere votata in tre modi possibili: SI, NO, NI (Non Interessante). Il risultato viene elaborato nel seguente modo: se il numero dei votanti che considerano la proposta non interessante è minore degli altri votanti, allora la proposta è considerata *accettata* o *rifiutata* a seconda che ci sia una maggioranza di SI o NO rispettivamente. In tutti gli altri casi la proposta viene considerata *ingiudicata*;

Si realizzi un'applicazione Java distribuita composta dalle classi *Proposta* e *Votante*. La classe *Proposta* viene istanziata con il contenuto della proposta, raccoglie le votazioni (anonime), elabora e restituisce il risultato. La classe *Votante* svolge le operazioni di trasmissione di istanze di *Proposta* tra un votante e il successivo, in accordo al protocollo di cui sopra.

Attenersi esattamente a quanto espresso dalle seguenti figure. Fig.1 mostra la configurazione di test con soli tre votanti. 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 proponente si distingue per avere in ingresso un parametro in più, che rappresenta il contenuto della proposta. Fig.5 mostra la sequenza di passi (ad alto livello) che le istanze di *Votante* svolgono in ogni caso di test. La logica di *Votante* e *Proposta* dovrà essere valida a prescindere dai numeri di porta e dal numero di votanti.

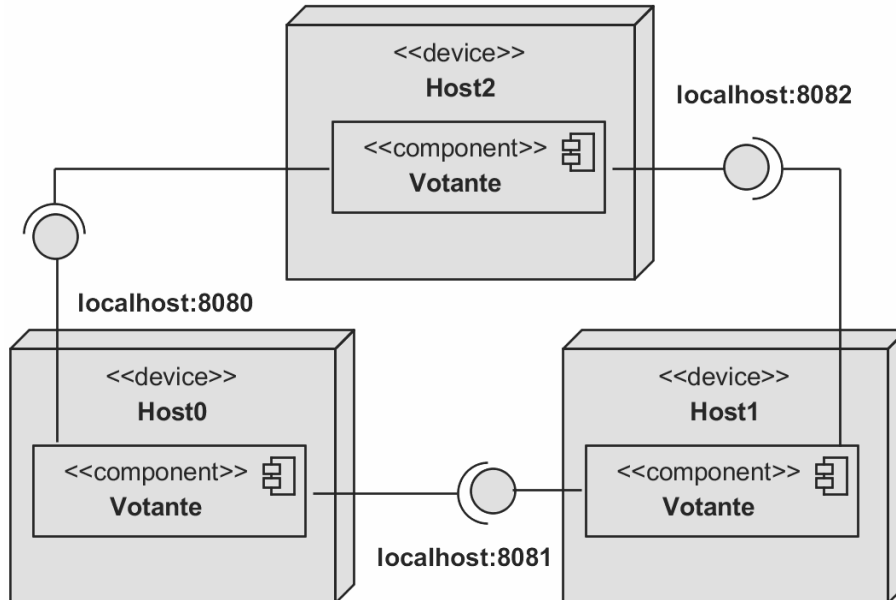


Fig.1 - Tre istanze di *Votante* e relative porte di ascolto

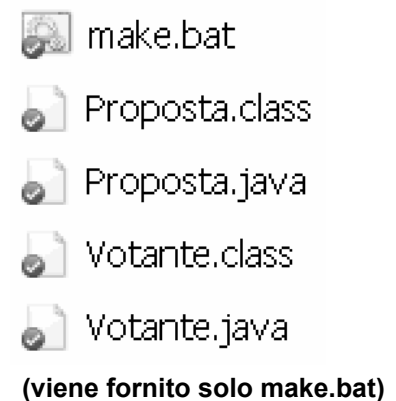


Fig.2 - c:\prg\esercizio2

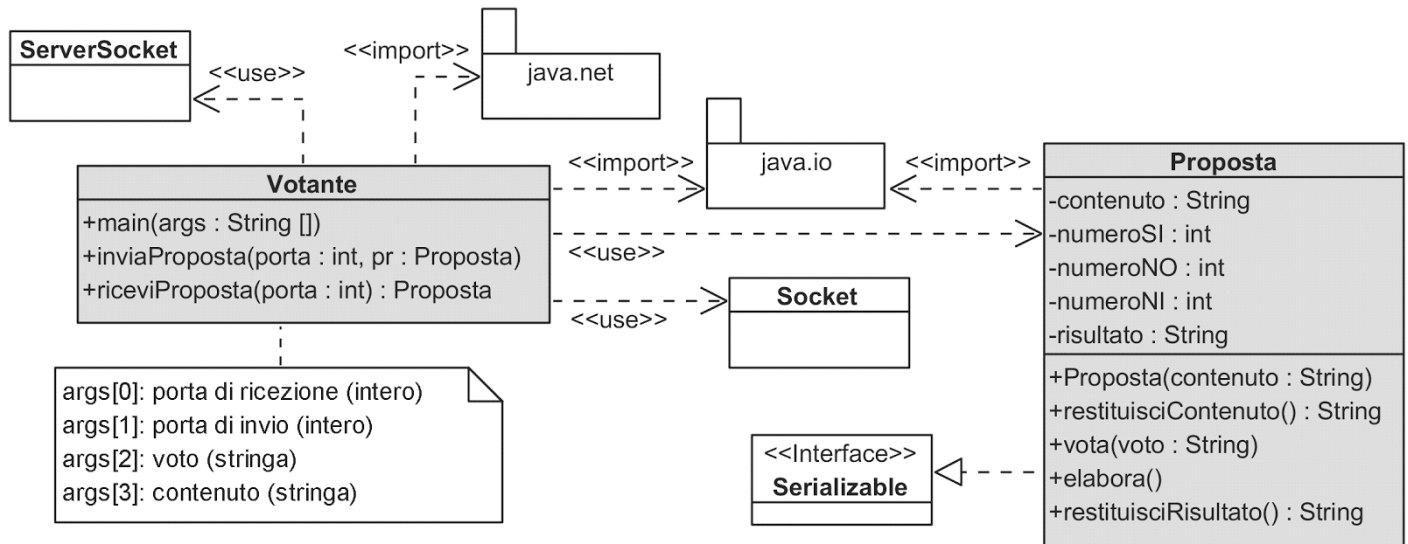


Fig.3 - Classi di cui deve essere composta l'applicazione (in grigio) e classi/package da usare (in bianco)

```

c:\prg\jdk8\bin\javac *.java
pause
start cmd /k "c:\prg\jdk8\bin\java Votante 8082 8080 SI"
pause
start cmd /k "c:\prg\jdk8\bin\java Votante 8081 8082 NO"
pause
start cmd /k "c:\prg\jdk8\bin\java Votante 8080 8081 SI "Propongo di introdurre gli OGM""
pause
rem come risultato stampa "accettata"
pause
start cmd /k "c:\prg\jdk8\bin\java Votante 8082 8080 NO"
pause
start cmd /k "c:\prg\jdk8\bin\java Votante 8081 8082 NO"
pause
start cmd /k "c:\prg\jdk8\bin\java Votante 8080 8081 SI "Propongo di abolire il limite di velocita""
pause
rem come risultato stampa "rifiutata"
pause
start cmd /k "c:\prg\jdk8\bin\java Votante 8082 8080 NI"
pause
start cmd /k "c:\prg\jdk8\bin\java Votante 8081 8082 NO"
pause
start cmd /k "c:\prg\jdk8\bin\java Votante 8080 8081 SI "Propongo di introdurre l'eutanasia""
pause
rem come risultato stampa "ingiudicata"

```

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

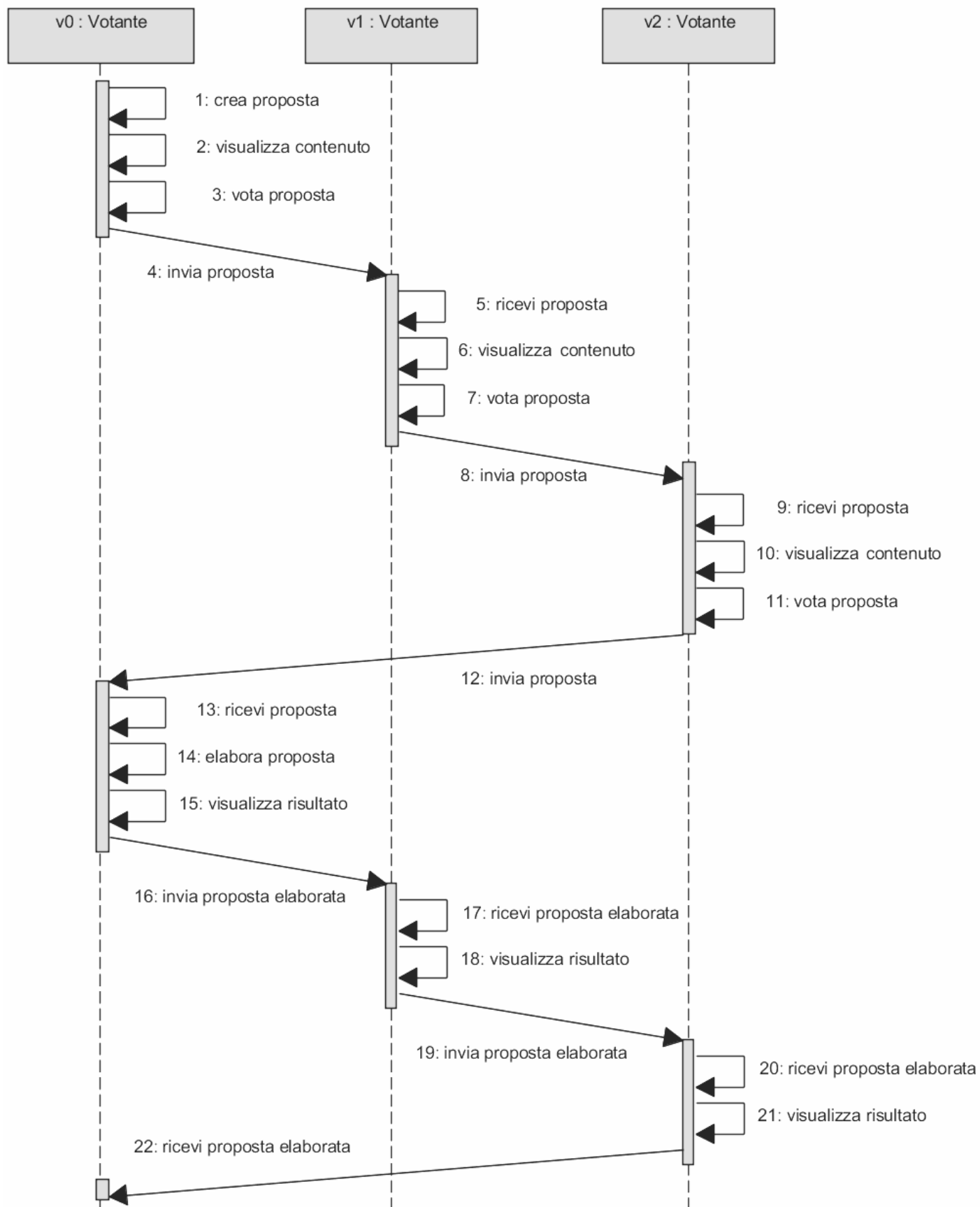


Fig.4 - Sequenza di passi che le istanze di *Votante* devono svolgere. Per brevità non viene rappresentata la classe *Proposta*.