

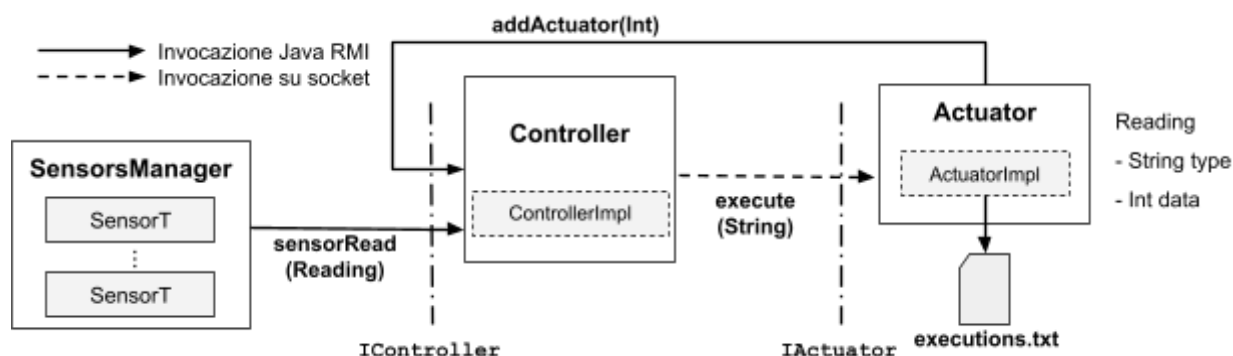
Università degli Studi di Napoli Federico II
Esame di Advanced Computer Programming
Proff. De Simone, Della Corte

Prova pratica del giorno 16/01/2024
Durata della prova: 120 minuti

*Lo studente legga attentamente il testo e produca il programma ed i casi di test necessari per dimostrarne il funzionamento. La mancata compilazione dell'elaborato, la compilazione con errori o l'esecuzione errata daranno luogo alla valutazione come **prova non superata**.*

Al termine della prova lo studente dovrà far verificare il funzionamento del programma ad un membro della Commissione.

Testo della prova



Il candidato realizzi un sistema per la gestione di sensori remoti (le cui letture sono rappresentate da istanze di **Reading**). Il sistema è composto dalle seguenti entità.

Controller. Espone i servizi Java RMI **addAcuator** e **sensorRead**. Il metodo **addActuator** consente l'aggiunta di un Actuator al Controller, ed è caratterizzato dal parametro **port** (int), che rappresenta la porta su cui è in ascolto l'Actuator. Il Controller mantiene un elenco degli Actuator registrati. Il metodo **sensorRead** consente l'invio di una lettura **Reading**. Per ogni invocazione del metodo, il Controller invoca ad uno ad uno gli Actuator disponibili nell'elenco, finché ne individua uno che non restituisca **false**. Se tutti gli Actuator restituiscono **false**, **sensorRead** restituirà **false**.

SensorsManager. Emula un insieme di sensori che genera dati. SensorsManager avvia 10 thread; ciascun thread invoca `sensorRead` su Controller passando un'istanza di `Reading`. Il valore del campo `type` (stringa) di `Request` è scelto a caso tra `temperature` e `pressure`; `data` (intero) è scelto a caso tra 1 e 50.

Actuator. Offre il metodo **execute(String)**, dove la stringa è la concatenazione dei due attributi di Reading. Ciascun Actuator può eseguire una sola **execute** in maniera concorrente. Se Actuator sta già eseguendo una richiesta, l'invocazione di **execute** termina immediatamente con esito **false**. In caso contrario, **execute** i) estrae il tipo di richiesta e il dato dalla stringa ricevuta, ii) scrive tali informazioni a video e su file (executions.txt) e iii) restituisce esito true. Una invocazione di **execute** dura un tempo scelto a caso tra 1 e 5 secondi.

Il sistema sarà testato da prompt con 1 SensorsManager, 1 Controller, ed un numero arbitrario di Actuator. Un Actuator sarà avviato come nel seguente esempio (porta e nome del file su cui saranno salvate le informazioni sono specificati da prompt):

```
java actuator.Actuator 8000 executions.txt
```

Il candidato utilizzi Java RMI per i metodi di Controller e proxy-skeleton con socket TCP per gli Actuator. A tal fine, il candidato predisponga le opportune interfacce e le classi Proxy-Skeleton. Si utilizzino inoltre lock per la mutua esclusione e skeleton per ereditarietà per gli Actuator.