

Risolutore di puzzle – Parte 3

Programmazione concorrente e distribuita
Progetto A.A. 2014/2015

1 Descrizione del progetto

La parte 3 del progetto consiste nel modificare il programma realizzato per la parte 2 in modo da implementare un sistema client-server con Java RMI. In particolare il client si deve occupare di gestire i file di input e di output – identici a quelli delle parti precedenti – e di inviare al server le informazioni che permettono a quest'ultimo di risolvere il puzzle.

Il server avrà un nome, specificato da linea di comando, che permette al client di connettersi a quest'ultimo; anche il client perciò dovrà accettare il nome del server a linea di comando. Il riferimento remoto all'oggetto client non può essere inserito nel registro pubblico RMI.

L'input e l'output del client rimangono identici: l'input consiste di un file contenente l'elenco completo dei pezzi mischiati, e l'output di un file contenente il testo del puzzle, il puzzle ricostruito in forma tabellare e le sue dimensioni.

I dettagli su come implementare la logica di comunicazione sono lasciati allo studente, che dovrà darne spiegazione nella relazione allegata. È richiesto inoltre che il progetto sia robusto, cioè che l'interruzione improvvisa della comunicazione client-server venga gestita in modo opportuno e che tale soluzione sia spiegata nella relazione.

2 Requisiti obbligatori

La consegna della parte 3 del progetto deve soddisfare i seguenti requisiti obbligatori:

- La classe contenente il metodo main del client si chiamerà `PuzzleSolverClient`, quella server `PuzzleSolverServer`
- Il client accetterà 3 parametri:
`nome_file_input nome_file_output nome_del_server`
- Il server accetterà 1 parametro:
`nome_del_server`
- il registro RMI conterrà solo il riferimento al server, non quello del client.
- L'input e l'output del client avranno le stesse specifiche definite nella parte 1 del progetto.
- Il client e il server non avranno un'interfaccia grafica.
- In allegato ci sarà una relazione.
- In allegato ci sarà un **makefile**.
- In allegato ai programmi ci saranno due **script Bash**, uno si chiamerà **puzzlesolver-client.sh** e l'altro **puzzlesolverserver.sh**. Gli script riceveranno gli stessi parametri delle rispettive classi Java.

3 Valutazione del progetto - Relazione

Al fine di permettere la valutazione del progetto è richiesto di consegnare una **breve relazione** che illustri in modo **preciso e sintetico**:

- la spiegazione della logica di comunicazione client-server;
- come viene gestita la robustezza del programma (gestione dell'interruzione improvvisa del client, del server, del canale di comunicazione tra client e server);
- i cambiamenti effettuati rispetto la parte 2: cosa è cambiato nel codice e nell'organizzazione delle classi e perché?

NOTA: La valutazione del progetto dipenderà in **ugual misura dalla qualità della relazione e dalla correttezza e qualità del codice**. Si osserva inoltre che la valutazione del progetto terrà conto della capacità di risolvere il problema assegnato in modo semplice ma esauriente. Sono quindi preferibili i progetti che implementano in modo semplice e chiaro una soluzione corretta, mentre viene **SCORAGGIATA l'aggiunta di funzionalità e aspetti grafici non richiesti dalla presente specifica**. Anche in questa fase la valutazione del codice terrà conto dell'aderenza ai principi basilari della programmazione ad oggetti in Java.

4 Regole per la consegna del progetto

Il progetto dovrà essere realizzato da ogni singolo studente in modo **INDIPENDENTE**.

4.1 Come verrà effettuato il test operativo del progetto

Per quanto riguarda la valutazione del progetto, questo verrà eseguito localmente su un sistema operativo Linux, dove vi è una installazione di Java 7. I computer dei laboratori attualmente hanno per default la versione 6 di Java, ma è possibile usare la versione 7 personalizzando l'ambiente di esecuzione come indicato alla pagina

<http://www.studenti.math.unipd.it/index.php?id=corsi#c620>.

Al fine di standardizzare la procedura di valutazione il programma dovrà aderire a quanto specificato nella sezione "Requisiti obbligatori".

ATTENZIONE: se un progetto non compila o non soddisfa i i requisiti obbligatori sarà considerato insufficiente.

4.2 Cosa consegnare

La cartella principale del pacchetto da consegnare, dovrà essere chiamata "parte3". Nel caso di consegna contemporanea di più parti, il pacchetto avrà più cartelle, una per ciascuna parte.

4.3 Come e quando consegnare

Il progetto va consegnato dalle macchine del laboratorio invocando il comando

```
consegna programmazione3-14-15
```

dalla directory contenente i file da consegnare. **Non** saranno accettate altre modalità di consegna (ad es. via email). È possibile consegnare remotamente il progetto usando il server `ssh.studenti.math.unipd.it`.

Le date di consegna del progetto saranno indicate sul sito del corso.