

SAPIENZA Università di Roma
Facoltà di Ingegneria dell'Informazione, Informatica e Statistica
Corsi di Laurea in Ingegneria Informatica ed Automatica ed in Ingegneria dei Sistemi Informatici
Corso di Progettazione del Software
Esame del **9 Giugno 2014**
Tempo a disposizione: 3 ore

Requisiti. L'applicazione da progettare riguarda una parte di un videogioco fantasy per tablet e smartphone. Un gioco è costituito da diversi giocatori, ciascuno dei quali appartiene esclusivamente al gioco stesso. I giocatori, che hanno un nome (una stringa) sono divisi in 4 categorie: i principi, le principesse, i maghi e i draghi. Ai principi possono piacere diverse principesse, ma possono essere innamorati solo di una tra queste. Analogamente alle principesse possono piacere diversi principi, ma possono essere innamorate di uno solo principe, tra quelli che piacciono loro. Le principesse hanno accesso a spade incantate che vengono date ai principi per compiere le loro imprese. Le spade possono rompere incantesimi di maghi. Ogni spada è accessibile da una sola principessa. Ogni mago può controllare al più un drago.

In questo compito siamo interessati al comportamento dei soli principi. Un principe è inizialmente alla *corte*. Quando è alla *corte*, se riceve un *bacio* da una principessa che gli piace, e non è già innamorato, se ne innamora rimanendo alla *corte*. Se invece *riceve una spada incantata e l'incarico di sconfiggere un drago* da una principessa (la spada deve essere accessibile alla principessa, che se ne priva dandola al principe), parte per l'*impresa*. Durante l'*impresa*, se il drago cercato lo *attacca* e il drago è controllato da un mago e la spada rompe i suoi incantesimi, allora vince tornando alla *corte*, mandando un *bacio* alla principessa che lo ha incaricato, e restituendogli la spada; altrimenti perde e viene trasformato in un *ranocchio*. Quando è un *ranocchio*, se riceve un *bacio* da una principessa, torna alla *corte* nelle sembianze di principe e si innamora della principessa (che, se non gli piaceva, ora gli piace).

Siamo interessati alla seguente attività principale. L'attività prende in input un gioco G e concorrentemente esegue le seguenti due sottoattività: (i) gioca, e (ii) analisi. La sottoattività di gioco (i) avvia il gioco attivando tutti i giocatori di G mandando opportuni eventi (i dettagli non interessano). Poi si mette in attesa del comando di fine-gioco da parte dell'utente che interrompe il gioco. La sottoattività di analisi (ii) calcola quanti principi, principesse, maghi e draghi sono presenti nel gioco G e per ciascuna categoria ne stampa il numero. Una volta che tali sottoattività sono state completate, si ricalcola il numero di principi e principesse del gioco G innamorate tra loro e lo stampa.

Domanda 1. Basandosi sui requisiti riportati sopra, effettuare l'analisi producendo lo schema concettuale in UML per l'applicazione, comprensivo del diagramma delle classi (inclusi vincoli non esprimibili in UML), diagramma stati e transizioni per la classe *Principe*, diagramma delle attività, specifica del diagramma stati e transizioni, e specifica dell'attività principale e delle sottoattività NON atomiche (indicando in modo esplicito quali attività atomiche sono di I/O e quali sono Task), motivando, qualora ce ne fosse bisogno, le scelte di progetto.

Domanda 2. Effettuare il progetto, illustrando i prodotti rilevanti di tale fase e motivando, qualora ce ne fosse bisogno, le scelte di progetto. È obbligatorio definire solo le responsabilità sulle associazioni del diagramma delle classi.

Domanda 3. Effettuare la realizzazione, producendo un programma JAVA e motivando, qualora ce ne fosse bisogno, le scelte di progetto. È obbligatorio realizzare in JAVA solo i seguenti aspetti dello schema concettuale:

- La classe **Principe** con classe **PrincipeFired**, eventuali superclassi, e le classi JAVA per rappresentare le *associazioni* di cui queste classi hanno responsabilità.
- L'*attività principale* e le sue eventuali sottoattività NON atomiche.