

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 **25 luglio 2016**
Tempo a disposizione: 3 ore

Requisiti. L'applicazione da progettare riguarda una parte di un videogioco su robottini programmabili per tablet e smartphone. Un gioco ha un nome (una stringa) e vi partecipano diversi giocatori. Ogni giocatore ha una insieme di robottini ciascuno con un nome e delle coordinate cartesiane (2 reali) che ne indicano la posizione. Ciascun robottino appartiene solo ad un giocatore. I robottini sono suddivisi in robottini bipedi e quadrupedi. Dei primi interessa il raggio di azione (cioè un reale), mentre dei secondi interessa la velocità di spostamento (un reale). Ogni giocatore ha almeno un robottino bipede che funge da coordinatore.

In questo compito siamo interessati al comportamento dei robottini bipedi. Un robottino bipede è inizialmente *in posizione*. Quando riceve un comando *nuova posizione* (non importa da chi) con payload delle coordinate, se le coordinate sono nel proprio raggio, cambia le proprie coordinate e ritorna *in posizione*, se invece le coordinate indicate sono fuori dal proprio raggio di azione, *chiede aiuto* ad un robottino quadrupede appartenente allo stesso giocatore (non interessa come scelto) per farsi portare alle coordinate richieste mettendosi in uno stato di *in viaggio*. Quando è *in viaggio* e riceve dal robottino quadrupede il segnale di *arrivo* aggiorna la propria posizione a quella indicata originariamente e si mette *in posizione*.

Siamo interessati alla seguente attività principale. L'attività prende come parametro di input un gioco G e concorrentemente esegue le seguenti due sottoattività: *Gioca*, e *Analizza*. La sottoattività *Gioca* 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 *Analizza* che calcola per ogni giocatore del gioco G quanti robottini bipedi e quadrupedi gli appartengono, segnalandolo in output. Una volta che tali sottoattività sono state completate, si calcola la lista di tutti i robottini appartenenti al gioco G con le proprie coordinate e la si segnala in output, terminando così l'attività principale.

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 *RobottinoBipede*, diagramma delle attività, specifica del diagramma stati e transizioni, specifica dell'attività principale e delle sottoattività NON atomiche, 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 *Robottino*, la sottoclasse *RobottinoBipede* con classe *RobottinoBipedeFired*, le classi JAVA per rappresentare le *associazioni* di cui queste classi hanno responsabilità.
- L'*attività principale* (NON le sue sottoattività *Gioca* e *Analizza*, nè le sue sottoattività atomiche).