

SAPIENZA Università di Roma
Facoltà di Ingegneria dell'Informazione, Informatica e Statistica

Esercitazioni di
PROGETTAZIONE DEL SOFTWARE
(Corso di Laurea in Ingegneria Informatica ed Automatica
Corso di Laurea in Ingegneria dei Sistemi Informatici)
A.A. 2010-11

Esercitazione: dall'analisi alla realizzazione

SOLUZIONE

Requisiti

L'applicazione da progettare riguarda la gestione di costruzioni per bambini. Sono di interesse le scatole di montaggio, ciascuna caratterizzata da una descrizione testuale e dai tipi di mattoncini che essa contiene (almeno uno) con le rispettive quantità. Alcune scatole sono speciali e sono caratterizzate da un livello di difficoltà (un intero). Tali scatole speciali contengono almeno un tipo di mattoncini elettrificati (per esempio motorini elettrici, dispositivi luminosi, ecc.). I tipi di mattoncini sono caratterizzati dalle dimensioni (rappresentate da una stringa) e dal colore (una stringa). I tipi di mattoncini elettrificati sono inoltre caratterizzati da una specifica elettrica (una stringa). Oltre alle scatole di montaggio sono di interesse le costruzioni realizzabili con i mattoncini. Ogni costruzione è caratterizzata dalle istruzioni di montaggio (una stringa) e dai tipi di mattoncini richiesti, con le rispettive quantità. Alcune costruzioni sono speciali in quanto servono a illustrare un fenomeno elettrico, queste sono caratterizzate da una descrizione testuale del fenomeno (una stringa) e contengono esattamente un tipo di mattoncini elettrificati (in quantità arbitraria).

Requisiti (cont.)

Il fruitore della applicazione è interessato ad effettuare diverse operazioni, in particolare:

- data una scatola di montaggio s ed una costruzione c , verificare se s contiene tutti i tipi di mattoncini richiesti per c in quantità sufficienti;
- dato un tipo di mattoncino elettrificato me , restituire l'insieme C delle costruzioni in cui me è richiesto.

Requisiti (cont.)

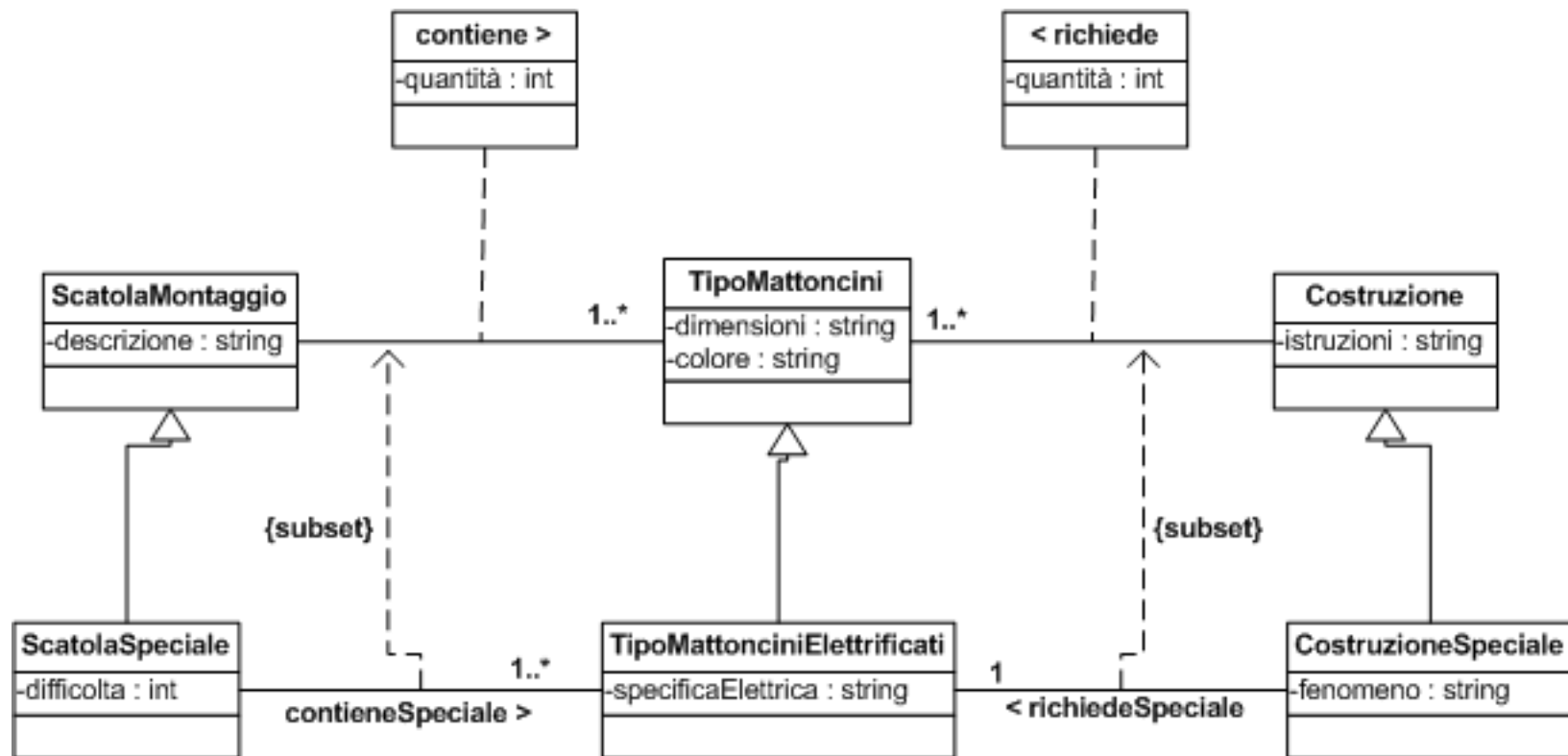
Domanda 1. Basandosi sui requisiti riportati sopra, effettuare la fase di analisi producendo lo schema concettuale in UML per l'applicazione e motivando, qualora ce ne fosse bisogno, le scelte effettuate.

Domanda 2. Effettuare la fase di progetto, illustrando i prodotti rilevanti di tale fase e motivando, qualora ce ne fosse bisogno, le scelte effettuate.

Domanda 3. Effettuare la fase di realizzazione, producendo un programma Java e motivando, qualora ce ne fosse bisogno, le scelte effettuate.

Fase di analisi

Diagramma delle classi



Fase di progetto

Responsabilità sulle associazioni

La seguente tabella delle responsabilità si evince da:

- 1. i requisiti,
- 2. la specifica degli algoritmi per le operazioni di classe e use-case,
- 3. i vincoli di molteplicità nel diagramma delle classi.

Associazione	Classe	ha resp.
<i>contiene</i>	<i>ScatolaMontaggio</i> <i>TipoMattoncini</i>	$\bar{S}I^{2,3}$ NO
<i>contieneSpeciale</i>	<i>ScatolaSpeciale</i> <i>TipoMattonciniElettrificati</i>	$\bar{S}I^3$ NO
<i>richiede</i>	<i>Costruzione</i> <i>TipoMattoncini</i>	$\bar{S}I^{2,3}$ $\bar{S}I^2$
<i>richiedeSpeciale</i>	<i>CostruzioneSpeciale</i> <i>TipoMattonciniElettrificati</i>	$\bar{S}I^3$ NO

Strutture di dati

Abbiamo la necessità di rappresentare collezioni omogenee di oggetti, a causa:

- dei vincoli di molteplicità $0..*$ delle associazioni,
- delle variabili necessarie per vari algoritmi.

Per fare ciò, utilizzeremo le classi del collection framework di Java: Set, HashSet.

Tabelle di gestione delle proprietà di classi UML

Riassumiamo le nostre scelte differenti da quelle di default mediante la *tabella delle proprietà immutabili* e la *tabella delle assunzioni sulla nascita*.

Classe UML	Proprietà immutabile
<i>ScatolaMontaggio</i>	<i>descrizione</i>
<i>ScatolaSpeciale</i>	<i>difficolta</i>
<i>TipoMattoncini</i>	<i>dimensioni</i>
	<i>colore</i>
<i>TipoMattonciniElettrificati</i>	<i>specificaElettrica</i>
<i>Costruzione</i>	<i>istruzioni</i>
<i>CostruzioneSpeciale</i>	<i>fenomeno</i>

Classe UML	Proprietà	
	nota alla nascita	non nota alla nascita
<i>CostruzioneSpeciale</i>	—	<i>richiedeSpeciale</i>

Altre considerazioni

Sequenza di nascita degli oggetti: Non dobbiamo assumere una particolare sequenza di nascita degli oggetti.

Valori alla nascita: Non sembra ragionevole assumere che per qualche proprietà esistano valori di default validi per tutti gli oggetti.