

Metodi Avanzati di Programmazione
Corso di Laurea in Informatica
Anno Accademico 2022/2023

Prova scritta del 19 Luglio 2023, Turno A

1) Scrivere le specifiche algebriche (semantiche e di restrizione) in **forma di equazioni** per il tipo astratto **Rete** di cui si forniscono le seguenti specifiche sintattiche:

Tipi:

Rete, Nodo, Peso, Intero, Booleano

Operatori:

crea()→Rete

aggiungiNodo(Rete,Nodo)→Rete // aggiunge un nodo alla rete

aggiungiNodoConnesso(Rete, Nodo, Nodo, Peso)→Rete // aggiunge un nodo (secondo parametro) alla rete e crea una connessione dal nodo aggiunto al nodo già presente nelle rete (terzo parametro) con peso pari a quello specificato (quarto parametro)

conta(Rete, Peso) → Intero //conta le connessioni nella rete con peso uguale a quello specificato

cancella(Rete, Nodo) →Rete //cancella il nodo specificato (secondo parametro) dalla rete solo se tale nodo è presente nella rete senza connessioni; solleva errore se il nodo non è presente nella rete; lascia la rete inalterata senza fare cancellazioni se il nodo da cancellare è presente nella rete, ma è coinvolto in connessioni

maggiore(Rete, Nodo, Nodo)→Nodo // restituisce il nodo della rete tra i due passati come parametri che ha numero più alto di archi in uscita nella rete; solleva errore se uno dei due nodi è assente dalla rete

uguale(Rete, Rete)→Booleano // restituisce vero se le due reti hanno uguale uguale somma dei pesi; falso altrimenti

N.B. Qualora si decida di svolgere l'esercizio in forma tabellare, scrivere almeno due equazioni in maniera esplicita. Dichiarare tutte le assunzioni fatte per svolgere l'esercizio.

(11 punti)

2) Descrivere in maniera **esaustiva** la relazione di ereditarietà nel paradigma OO con esempi in UML. Progettare un diagramma delle classi che includa la classe **Libro** con campi **titolo**, **autore** ed **prezzo**, costruttore, e metodo **toString()** che restituisce una stringa che rappresenta **titolo**, **autore** ed **prezzo**; la classe **Disciplina** con campi **settore** e **nroCFU**, costruttore, e metodo **toString()** che restituisce una stringa che rappresenta **settore** e **nroCFU**; la classe **LibroScolastico** che estende **Libro** e **Disciplina**, include il campo **corsoDiLaurea**, costruttore, e il metodo **toString** che restituisce una stringa che rappresenta **titolo**, **autore**, **prezzo**, **settore**, **nroCFU** e **corsoDiLaurea**. Chiarire le forme di ereditarietà usate nel progetto e commentarne vantaggi e limiti qualora si dovesse tradurre il progetto in Java.

(11 punti)

3) Descrivere in **maniera esaustiva** la gestione delle eccezioni in Java. Modellare in Java la classe **Chiave** con campo **id** di tipo String, **Valore** con campo **val** di tipo String, la classe **Dizionario** (scegliendo opportunamente il contenitore tenendo conto che si deve implementare i metodi **crea()**→Dizionario, **aggiungi**(Chiave, Valore)→Dizionario, **appartiene**(Dizionario, Chiave)→Booleano, **toString**(Dizionario)→String). Per il metodo **toString** usare opportunamente l'astrazione di controllo per sollevare una eccezione controllata **DizionarioVuoto** quando si prova a verificare l'elemento di una chiave a un dizionario vuoto e mostrare come questa può essere gestita. Mostrarne l'uso di questa classe e dei suoi servizi.

Gli iscritti che avrebbero dovuto frequentare l'esame di MAP negli AA 2017-2018, 2018-2019, 2019-2020 o 2020-2021 (perché allora iscritti al secondo anno) DEVONO rispondere alla domanda:

Usando la programmazione funzionale scrivere il metodo che, in **Dizionario**, realizza l'operatore **conta**(Dizionario, Valore)→Intero che conta le occorrenze di valore nel dizionario. Commentare le scelte fatte.

Gli iscritti che avrebbero dovuto frequentare l'esame di MAP negli AA precedenti al 2017-2018 possono rispondere alla domanda precedente o in alternativa alla seguente domanda:

Descrivere in maniera esaustiva classi astratte e interfacce nel paradigma OO con esempi in UML e mostrare analogie e differenze tra questi due modelli concettuali.

(11 punti)