

Metodi Avanzati di Programmazione, Corso di Laurea in Informatica, AA 2018-2019

Prova in Itinere 16 Aprile 2019, ore 14:00 – Traccia A

1) Fornire il **set completo e minimale di equazioni** per le specifiche algebriche (semantiche e di restrizione) del tipo astratto “**Insieme**” di cui si forniscono le specifiche sintattiche:

Tipi: Insieme, Stringa, Intero, Booleano

Operatori:

creaInsieme() → Insieme

aggiungi(Insieme, Stringa) → Insieme

cancella(Insieme, Stringa) → Insieme

cardinalità(Insieme) → Intero

filtra(Insieme, Stringa) → Insieme // rimuove tutte le stringhe dell’insieme che hanno come prefisso la stringa specificata ; restituisce errore se non si è riuscito a rimuovere alcuna stringa

uguale(Insieme, Insieme) → Booleano // restituisce vero se i due insiemi contengono gli stessi elementi , falso altrimenti

unione(Insieme, Insieme) → Insieme // restituisce l’unione dei due insiemi

[11 punti]

2) Descrivere in maniera esaustiva la astrazione di funzione (anche tramite esempi) e spiegare il rapporto di questa con la astrazione funzionale. **[6 punti]**

3) Descrivere in maniera esaustiva la relazione is_a nel paradigma OO. Formulare esempi usando UML. **[8 punti]**

4) Descrivere la differenza tra ereditarietà singola ed ereditarietà multipla sia in riferimento alle interfacce sia in riferimento alle classi e mostrare eventuali problemi. Formulare esempi in UML. Mostrare come simulare la ereditarietà multipla tra classi qualora questa non fosse supportata. Commentare i limiti della soluzione simulata. **[8 punti]**

Metodi Avanzati di Programmazione, Corso di Laurea in Informatica, AA 2018-2019

Prova in Itinere 16 Aprile 2019, ore 14:00 – Traccia B

1) Fornire il **set completo e minimale di equazioni** per le specifiche algebriche (semantiche e di restrizione) del tipo astratto “**Insieme**” di cui si forniscono le specifiche sintattiche:

Tipi: Insieme, Stringa, Intero, Booleano

Operatori:

creaInsieme() → Insieme

aggiungi(Insieme, Stringa) → Insieme

cancella(Insieme, Stringa) → Insieme

cardinalità(Insieme) → intero

filtra(Insieme, Intero) → Insieme // rimuove tutte le stringhe dell’insieme che hanno lunghezza maggiore dell’intero specificato ; restituisce errore se non è riuscito a rimuovere alcuna stringa

contenuto(Insieme, Insieme) → Booleano // restituisce vero se il primo insieme è contenuto nel secondo, falso altrimenti

intersezione(Insieme, Insieme) → Insieme // restituisce l’intersezione dei due insiemi

[11 punti]

2) Descrivere in maniera esaustiva la astrazione di procedura (anche tramite esempi) e spiegare il rapporto di questa con la astrazione funzionale. **[6 punti]**

3) Descrivere in maniera esaustiva il polimorfismo per inclusione nel paradigma OO. Formulare esempi di quanto descritto con riferimento a C++ e Java **[8 punti]**

4) Descrivere in maniera esaustiva la ereditarietà per implementazione fornendo anche esempi in UML. Mostrare come simulare la ereditarietà per implementazione qualora questa non fosse supportata. Commentare i limiti della soluzione simulata. **[8 punti]**