

Corso di Programmazione

Esame del 16 Luglio 2004

cognome e nome

Risolvi i seguenti esercizi e riporta le soluzioni in modo chiaro, giustificandole sinteticamente.

1. Procedure in Scheme

In questo esercizio le frazioni sono rappresentate da coppie di numeri interi (p, q) dove $p \geq 0$ e $q > 0$. Una coppia (frazione) si dice in *forma normale* se p e q sono primi fra loro. Se le coppie (p, q) e (p', q') rappresentano la stessa frazione, cioè se $p/q = p'/q'$, e se (p', q') è in forma normale, allora (p', q') è la normalizzazione di (p, q) .

Definisci un programma in Scheme che, data una lista di coppie che rappresentano frazioni, permetta di calcolare la lista delle rispettive normalizzazioni, preservandone l'ordine.

2. Scheme e Java

Traduci la seguente procedura Scheme in un corrispondente metodo statico nel linguaggio Java, basato sulla stessa struttura ricorsiva.

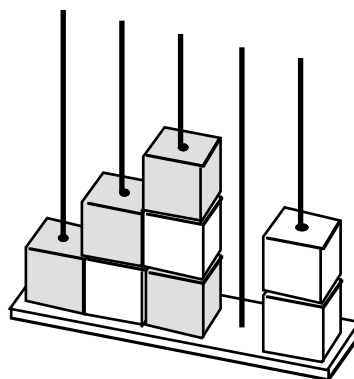
```
(define gf
  (lambda (n)
    (if (= n 1)
        1
        (- (* 2 (+ (gf (quotient n 2)) (remainder n 2))) 1)
    )))
```

3. Programmazione dinamica

Realizza un programma iterativo in Java che applichi la tecnica di programmazione dinamica per trasformare la soluzione ricorsiva dell'esercizio precedente.

4. Classi in Java

La classe *Tris* permette di istanziare modelli di un gioco che si ispira a *forza 4* e che può essere descritto come segue. Due giocatori hanno a disposizione dei cubetti forati, di colore rosso o di colore giallo il secondo, che a turno infilano su una delle n asticelle verticali (sopra gli altri se ve ne sono), cercando di allinearne 3 del proprio colore verticalmente, orizzontalmente o diagonalmente; vince chi vi riesce per primo. La figura illustra un esempio di dimensione $n = 5$ in cui il colore rosso (più scuro) ha vinto. In ogni caso, non si possono disporre più di n cubetti sulla stessa asticella.



```
public class Tris {
    public static final int RED = 0, YELLOW = 1; // costanti pubbliche per
    ... .. // rappresentare i colori

    public Tris(int n) /* n >= 3 */ { ... }
    public void play(int color, int bar) { ... }
    public boolean success(int color) { ... }
}
```

Le specifiche relative al protocollo della classe *Tris* sono le seguenti: il costruttore crea la configurazione iniziale del gioco di dimensione n , in cui le n asticelle sono libere; il metodo *play(color, bar)* permette di disporre un cubetto di colore *color* sull'asticella *bar*; *success(color)* permette di verificare se il giocatore con i cubetti di colore *color* ha vinto.

Proponi una realizzazione della classe *Tris*, formalizzando in Java lo stato interno e il corpo del costruttore e di ciascuno dei metodi.

5. Invarianti di classe

Proponi opportuni invarianti della classe *Tris* realizzata nell'esercizio precedente, rappresentandoli in notazione matematica oppure nel linguaggio *Jass*.