FI2 (12 CFU) - Compito di esame del 14 settembre 2011 – a.a. 2010-11, *tempo a disposizione: 90 minuti*[1]

Problema 1

Si considerino i seguenti metodi:

```
static int boh(int[] a, int i, int j) {
  int m = a[i];
  if(i==j) return m;
  return Math.max(m, boh(a, i+1,j));
}

static int mistero(int[] a, int i, int j) {
  if(j-i <= 3) return boh(a, i, j);
  int s = (j-i+1)/3;
  return Math.max(mistero(a, i, i+s-1), Math.max(mistero(a, i+s, i+2*s-1), mistero(a, i+2*s, j)));
}

static int mistero(int[] a) {
  return mistero(a, 0, a.length-1);
}

static int boh(int[] a) {
  return boh(a, 0, a.length-1);
}
```

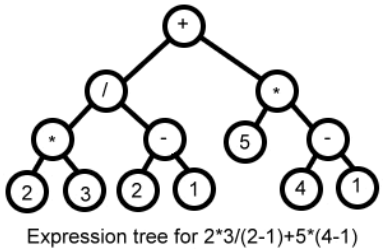
(a) Calcolare, motivandolo adeguatamente, il costo computazionale del metodo `mistero(int[])` in funzione della dimensione dell'input.

(b) Calcolare, motivandolo adeguatamente, il costo computazionale del metodo `boh(int[])` in funzione della dimensione dell'input.

(c) Riscrivere il metodo `mistero` senza utilizzare la ricorsione.

Problema 2

Un *expression tree* è un albero binario ai cui nodi interni è associato un operatore e alle cui foglie sono associati valori numerici (`double`). Ogni expression tree rappresenta dunque una espressione matematica, di cui può interessare valutare il risultato. Consideriamo in questo problema solo il caso di operatori binari (ogni nodo interno ha due figli). La figura illustra un esempio.



Expression tree for 2*3/(2-1)+5*(4-1)

(a) (Peso: 1/3) Definire le classi `ExpressionTree` e `ExpressionTreeNode` (solo variabili membro, costruttori e firme dei metodi rilevanti).

(b) (Peso: 2/3) Definire il metodo `double eval()` che calcola, in tempo lineare nel numero di operandi, il valore dell'espressione rappresentata da `this ExpressionTree` (operazioni possibili: somma, sottrazione, prodotto e divisione).

Problema 3

Si considerino i seguenti quesiti, relativi al concetto di grafo orientato (o diretto, o digrafo).

(a) Realizzare la classe `Digraph` (e le eventuali classi relative ad archi e nodi) che rappresenta un digrafo mediante liste di adiacenza. Le classi dovranno contenere, oltre alle variabili di istanza, anche le firme di tutti i metodi pubblici che ritenuti fondamentali per risolvere i problemi successivi.

(b) Scrivere un metodo Java che effettui la visita in profondità di un digrafo, etichettando nodi ed archi in maniera appropriata. Spiegare il significato delle quattro categorie di archi ("discovery", "back", "forward" e "cross") che vengono individuate nella visita e indicare il costo dell'algoritmo, motivandolo adeguatamente.

(c) Modificare il metodo scritto al punto (b) in modo che, durante la visita, vengano costruiti quattro nuovi grafi, rispettivamente isomorfi[2] ai sottografi individuati dagli archi "discovery", "back", "forward" e "cross", rispettivamente.

---

[1] Compilare i campi nel retro del foglio e seguire le istruzioni indicate per la consegna.

[2] Due grafi si dicono isomorfi se è possibile definire una corrispondenza biunivoca f fra i loro vertici in modo che esista l'arco (u, v) se e solo se esiste l'arco $(f(u), f(v))$.

Consegna dell'elaborato

1) Compilare i campi con le proprie informazioni

2) Piegare lungo la linea verticale ed utilizzare il foglio a meta' per contenere l'elaborato.

(Questa parte va dietro)

Compito d'esame - Fondamenti di Informatica II (FI2)

Matricola

Nominativo

Esame a cui si partecipa
Fondamenti di Informatica II