FI2 (12 CFU), ASD Nettuno (6 CFU) - Compito di esame del 24-7-2012 – a.a. 2011-12
tempo a disposizione: 90 minuti

Problema 1

Si considerino i seguenti metodi:

```
static void m(int[] a, int i) {
    if(i == a.length-1) {
        a[0] = a[i];
        return;
    }
    int tmp = a[i];
    m(a, i+1);
    a[a.length-1-i] = tmp;
}

static void m(int[] a) {
    m(a, 0);
}
```

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

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

(c) Riscrivere il metodo `m(int[])` senza far uso della ricorsione.

Problema 2

Con riferimento agli alberi binari generici, risolvere i punti seguenti.

(a) Definire due classi Java, `BinTree` e `Node`, per rappresentare alberi binari generici e per risolvere i successivi punti. Le classi debbono specificare variabili membro e costruttori; per quanto riguarda i metodi, sarà sufficiente l'indicazione delle loro firme.

(b) Dato un albero binario, definire un metodo Java `boolean isCompleto()` che determini se l'albero fornito è un albero completo oppure no. Determinare il costo computazionale del metodo.

(c) Dato un albero binario τ, definire un metodo Java `Node maxCompleto()` che restituisca la radice del sottoalbero completo più grande (avente più nodi) contenuto in τ. In caso di più sottoalberi completi aventi la stessa dimensione, restituirne uno qualsiasi. Determinare il costo computazionale del metodo.

Problema 3

Svolgere i punti seguenti.

(a) Definire classi Java utili a rappresentare un grafo non diretto (solo variabili membro, costruttori e firme dei metodi).

(b) Definire classi Java utili a rappresentare un grafo diretto (solo variabili membro, costruttori e firme dei metodi) e a supportare lo svolgimento del successivo punto (c).

(c) Con riferimento al caso (b), definire un metodo Java che esegua una Depth First Search che etichetti opportunamente gli archi.