

Programmazione

Appello del 6/07/2011

Esercizio 1 (8 punti)

- a. (4 punti) Scrivere l'output del seguente programma Java

```
public class Main {
public static void main(String[] args) {
    int [] A = {55,83,21,13,43,10,56};
    stampa(A);
    enigma(A);
    stampa(A);
}

public static void stampa (int [] A ){
    for (int x:A) {
        System.out.print (x + " ");
    }
    System.out.println();
}

public static void enigma (int A[]){
int i=0;
    while (i < A.length - 1){
        if (A[i] < A[i+1]) A[i+1]=A[i]-1;
        else A[i]=A[i+1]+1;
        i++;
    }
}
}
```

- b. (4 punti) Riscrivere la procedura enigma in modo equivalente senza far uso né del costrutto while né del costrutto for.

Esercizio 2 (7 punti)

Scrivere un metodo di classe *identità* che prende in input un intero **n** e restituisce la matrice identità (che deve essere creata dal metodo) di ordine n (una matrice n*n in cui tutti gli elementi sulla diagonale contengono 1 e tutti gli altri elementi 0).

n deve essere maggiore di 0, altrimenti il metodo restituisce **null**.

Esercizio 3 (12 punti)

- Scrivere in Java una classe Memoria che simuli la memoria di un calcolatore, organizzata come un array di B byte (B è un **int** preso in input dal **costruttore** della classe che deve essere l'unico presente). Impostare tutti i campi nella classe in modo opportuno in modo che non sia possibile manipolare la memoria dall'esterno se non attraverso i metodi pubblici che seguono.
- Aggiungere a Memoria un metodo **private int freeBlock(int length)** che restituisce la posizione iniziale di un blocco consecutivo di *length* locazioni libere. Se tale blocco non è presente, viene ritornato il valore convenzionale **-1**.
- Aggiungere a Memoria un metodo **public int set(byte [] values)** che memorizza consecutivamente in Memoria tutti i valori presenti nell'array *values*. Il metodo sfrutta il metodo privato **freeBlock** e ritorna la locazione in Memoria del primo valore memorizzato; viene sollevata un'opportuna eccezione se non c'è sufficiente spazio libero per la memorizzazione.
- Aggiungere a Memoria un metodo **public byte[] get(int start, int length)** che restituisce sottoforma di array i primi *length* valori memorizzati in memoria a partire dalla locazione *start*. Qualora almeno una di tali posizioni sia non assegnata, oppure inesistente, viene sollevata un'opportuna eccezione **java.lang.IndexOutOfBoundsException**.
- Aggiungere a Memoria un metodo **public void reset(int start, int length)** che elimina il contenuto di *length* posizioni a partire dalla posizione *start*.

*[si consiglia di aggiungere come campo della classe un array di **booleani** di dimensione B che tenga conto delle posizioni libere ed occupate della memoria.]*

Esercizio 4 (9 punti)

- Si scriva una classe astratta **Figura** per rappresentare figure piane
- La classe deve avere un metodo **public abstract double area ();** che sarà implementato dalle classi che estendono **Figura** con lo scopo di calcolare l'area della figura.
- La classe deve avere un metodo **public abstract double perimetro ();** che sarà implementato dalle classi che estendono **Figura** con lo scopo di calcolare il perimetro della figura.
- La classe deve inoltre implementare un metodo **public boolean minore (Figura f)** che ritorna **true** se e solo se sia l'area che il perimetro della figura su cui è invocato sono minori dell'area e del perimetro della figura **f**, rispettivamente. *[si sfruttino i metodi **area** e **perimetro**]*
- *Si implementino infine le seguenti classi **concrete** che implementano **Figura**, in ognuna delle quali bisogna implementare un **costruttore**, il metodo **area** e il metodo **perimetro**:*
 - La classe **Quadrato**, il cui costruttore prende un double (x) che rappresenta la lunghezza del lato del quadrato.
 - La classe **Triangolo Rettangolo**, il cui costruttore prende tre double (x,y,z) che rappresentano la lunghezza dei tre lati di un triangolo rettangolo.

Attenzione:

- Per svolgere il compito si hanno a disposizione **120** minuti.
- Scrivere **subito** nome, cognome, matricola e numero del compito su OGNI FOGLIO.
- Durante la prova scritta non è possibile abbandonare l'aula.
- Non è ammesso **per nessun motivo** comunicare in qualsiasi modo con altre persone
- **Non** è possibile consultare appunti, libri, dispense o qualsiasi altro materiale.
- Qualsiasi strumento elettronico di calcolo o comunicazione (telefoni cellulari, calcolatrici, palmari, computer, etc...) deve essere **completamente disattivato** e **depositato in vista sulla cattedra**