

Programmazione

Appello del 22/06/2011

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]=A[i+1];
        else A[i+1]=A[i];
        i++;
    }
}
}
```

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

Esercizio 2 (7 punti)

Scrivere un metodo di classe **differenza** che prende in input due matrici A e B di numeri interi, e restituisce in output la matrice A-B (che deve essere creata dal metodo). Il metodo deve controllare che le matrici A e B siano rettangolari ed abbiano le stesse dimensioni, altrimenti solleva un'opportuna eccezione.

Esercizio 3 (7 punti)

Scrivere un metodo di classe **reverse_search** che prende in input due array A e B di int, e restituisce l'intero x se l'array B ha lunghezza minore o uguale dell'array A ed è possibile individuare un sottoarray di A di elementi di posizione consecutiva, a partire dalla posizione x di A, uguale all'array B letto al contrario, -1 altrimenti.

Per esempio, se A={70, 1, 55, 83, 21, 13, 43, 10, 56}; e B={43, 13, 21}; deve essere ritornato **4**.

Esercizio 4 (6 punti)

Si considerino le seguenti classi.

```
class A{
    {
        stampa();
    }
    int x=9;
    {
        stampa();
    }
    public A(){
        x=10;
        stampa();
    }

    public A(int x){
        this();
    }
}
```

```
        this.x=x*2;
        stampa();
    }

    private void stampa(){
        System.out.println(x);
    }
}

class B extends A{
    public B(int x){
        super(3*x);
    }
}
```

Si dica cosa viene stampato a video dal seguente codice, segnalando eventualmente eventuali errori a tempo di esecuzione:

```
A a1 = new A();
System.out.println ("FINE");
A a2 = new A(2);
System.out.println ("FINE");
B b = new B(7);
```

Esercizio 5 (8 punti)

- Si scriva una classe astratta **Operazione** per rappresentare semplici operazioni tra numeri interi
- La classe deve avere un metodo **public abstract int risultato ();** che sarà implementato dalle classi che estendono **Operazione** con lo scopo di calcolare il risultato dell'operazione.
- La classe deve inoltre sovrascrivere il metodo **toString** in modo che stampi il risultato dell'operazione. *[si sfrutti il metodo risultato]*
- *Si implementino infine le seguenti classi concrete che estendono Operazione*, in ognuna delle quali bisogna inserire le necessarie **variabili di istanza** ed implementare un **costruttore** e il metodo **risultato**:
 - La classe **addizione**, il cui costruttore prende i due interi da addizionare.
 - La classe **sottrazione**, il cui costruttore prende i due interi da sottrarre.
 - La classe **moltiplicazione**, il cui costruttore prende i due interi da moltiplicare.

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**