

Cognome e Nome _____

Matricola _____

Programmazione 1

A.A. 2015/2016

Appello del 14 Giugno 2016

Esercizio 1 (8 punti)

1.1 (4 punti) Cosa stampa il seguente frammento di codice Java?

```
int [] A = {10,2,30,50,20};
for (int i=A.length-1;i>0;i--) {
    if (A[i]>A[i-1]){
        System.out.println(A[i-1]-A[i]);
    }
}
```

1.2 (4 punti) Cosa stampa il seguente programma Java?

```
public class Main {
    public static void main(String[] args) {
        System.out.println(enigma(5));
        System.out.println(enigma(11));
        System.out.println(enigma(1000));
    }

    static int enigma (int x){
        if (x<=0) return 0;
        return 3 + enigma (x-1);
    }
}
```

Esercizio 2 (12 punti)

- (4 punti) Scrivere un metodo **static int contaOccorrenze (int[] a, int n)** che, presi come parametri un array **a** di numeri interi e un intero **n**, restituisce il numero di occorrenze di **n** all'interno di **a**. Se **a** vale *null*, viene restituito 0. Ad esempio, se **a**={1, 4, 3, -7, 4, 2, 4, 1, 5} e **n**=1 viene restituito 2.
- (4 punti) Scrivere un metodo **static int massimeOccorrenze (int[] a)** che, preso come parametro un array **a** di numeri interi, sfruttando il metodo *contaOccorrenze*, restituisce il numero **massimo** di ripetizioni di uno stesso valore all'interno di **a**. Se **a** vale *null*, viene restituito 0. Ad esempio, se **a**={1, 4, 3, -7, 4, 2, 4, 1, 5} viene restituito 3 in quanto l'elemento più ripetuto (4) è ripetuto 3 volte.
- (4 punti) Scrivere un metodo **static int[] elementiPiuRicorrenti (int[] a)** che, preso come parametro un array **a** di numeri interi, sfruttando i metodi *contaOccorrenze* e *massimeOccorrenze*, restituisce un nuovo array dell'opportuna lunghezza contenente tutti e soli i valori dell'array **a** che occorrono il più alto numero di volte rispetto a tutti i valori presenti nell'array **a**. Se **a** vale *null*, viene restituito *null*. Ad esempio, se **a**={1, 4, 3, -7, 4, 1} e viene restituito {1, 4} (oppure {4,1}), in quanto 1 e 4 sono gli elementi di **a** che occorrono il maggior numero di volte (2 in questo esempio).

Esercizio 3 (6 punti)

Si consideri il tipo di dato

```
class Durata {
    int hh; //le ore
    int mm; //i minuti
    int ss; //i secondi
}
```

che rappresenta una durata (i minuti ed i secondi variano tra 0 a 59).

- **(3 punti)** Scrivere un metodo **static boolean isValid (Durata d)** che, presa come parametro una durata **d**, restituisce *true* se e solo se la durata è valida, ovvero contiene nei campi numeri non negativi, con il vincolo che *mm* e *ss* devono essere compresi tra 0 e 59. Se **d** vale *null*, viene restituito *false*.
- **(3 punti)** Scrivere un metodo iterativo **static void rendiValida (Durata d)** che, presa come parametro una durata **d**, nel caso **d** non sia valida la modifica facendo in modo che dopo la modifica **d** diventi una data valida nel modo seguente: se **d** conteneva almeno un campo negativo, la nuova durata sarà pari a 0 ore, 0 minuti e 0 secondi; se **d** conteneva minuti e/o secondi maggiori di 59, la nuova durata redistribuisce ore, minuti e secondi in modo da non alterare la durata totale. Se ad esempio **d** è 1 ora, 63 minuti e 150 secondi, viene trasformata in 2 ore, 5 minuti e 30 secondi.

Esercizio 4 (6 punti) [non è possibile utilizzare nessun metodo scritto a lezione senza riscriverlo completamente]

Si consideri il seguente tipo di dati visto a lezione, che rappresenta una lista di numeri interi positivi, **identificata dal suo primo elemento**.

```
class Elem {
    int valore;
    Elem next;
}
```

- **(3 punti)** Scrivere un metodo iterativo **public static int minimo (Elem lista)** che presa come parametro una lista di interi, restituisce il valore del più piccolo elemento presente in lista. Se la lista non esiste viene restituito **0**. Ad esempio se la lista è 3→5→6→2→6→4, viene restituito 2.
- **(3 punti)** Scrivere un metodo ricorsivo **public static int somma (Elem lista)** che presa come parametro una lista di interi, restituisce la somma di tutti gli elementi presenti nella lista. Se la lista non esiste, viene restituito **0**.

Regole per lo svolgimento della prova scritta:

- Per svolgere il compito si hanno a disposizione **90** minuti.
- Scrivere **subito** nome, cognome, matricola e numero del compito su **OGNI** FOGLIO.
- Le risposte al primo esercizio devono essere date direttamente nei riquadri di questo 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**
- Mettere in vista sul banco il proprio libretto o altro documento di identità.