Cognome e Nome _____

Matricola _____

Programmazione e Algoritmi I A.A. 2018/2019

Appello del 4 giugno 2019

Esercizio 1 (7 punti)

Si consideri il seguente programma Java:

```
public class MainClass {
      public static void main(String[] args) {
            int[] a = {10, 20, 2, 10, 9, 10, 2};
            System.out.println(enigma(a));
      }

      static int enigma(int[] a) {
            int ris = 0;
            for (int i=0; i<a.length; i++) {
                  boolean flag=false;
                  for (int j=0; j<i; j++) {
                        if (a[j] == a[i]) {
                              flag=true;
                        }
                  }
                  if (!flag) {
                        System.out.println
                            ("a[" + i + "]=" + a[i]);
                        ris++;
                  }
            }
            return ris;
      }

}
```

1.1 (4 punti) Cosa stampa il programma?

1.2 (3 punti) Si analizzi la complessità computazionale del metodo `enigma`.

Esercizio 2 (10 punti)

Scrivere un metodo in Java

`static boolean equivalenti (int[] a, int[] b)`

che, presi come parametro due array **a** e **b** di numeri interi, restituisce *true* se e solo se a e b contengono gli stessi elementi (non tenendo in conto le ripetizioni degli elementi e/o l'ordine in cui essi compaiono).

Se a e b valgono entrambi *null*, viene restituito *true*; se solo uno dei due array vale *null*, viene restituito *false*.

Ad esempio, se a={1, 8, 3, 1, 5, 10, 1, 4} e b={8, 1, 5, 10, 4, 3, 3}, il metodo deve restituire *true*.

Si analizzi la complessità temporale del metodo proposto: tale metodo può richiamare qualsiasi algoritmo di ordinamento e/o di ricerca visto a lezione e deve avere complessità temporale **O(n log n)** nel caso peggiore (*soluzioni con complessità temporale peggiore danno luogo a una valutazione minore, pari a un massimo di 5 punti*), dove n è la lunghezza dell'array a in input.

Esercizio 3 (8 punti)

Si consideri il seguente array di numeri interi:
{10, 20, 5, 50, 31, 14, 2, 6, 1, 8, 25}
Mostrare **passo-passo l'esecuzione dell'heap-sort** per ordinare l'array in modo non decrescente (compresa la fase iniziale di buildMaxHeap).

Esercizio 4 (10 punti)

Si considerino i seguenti tipi riferimento definiti dall'utente

```
class Esame {
      int codice;
      int cfu;
      int voto;
}
```

```
class Libretto{
      int matricola;
      Esame[] esamiSostenuti;
}
```

- **(5 punti)** Scrivere un metodo
  `static double mediaPonderata (Libretto l)`
  che, preso come parametro un libretto **l**, restituisce la media ponderata (rispetto ai CFU) degli esami presenti in **l.** Si ricorda che la media ponderata si calcola sommando *tutti i prodotti dei voti per i rispettivi CFU* e dividendo la somma così ottenuta per il numero totale di CFU.
  Il metodo deve gestire opportunamente tutti i casi in cui le variabili di tipo riferimento coinvolte valgano null.

- **(5 punti)** Scrivere un metodo
  `static int[] esamiSostenutiVotoAlmeno (Libretto l, int voto)`
  che, preso come parametro un libretto **l** e un intero **voto**, restituisce un array di interi contenente tutti e soli i codici degli esami sostenuti con voto maggiore o uguale a voto.
  Il metodo deve gestire opportunamente tutti i casi in cui le variabili di tipo riferimento coinvolte valgano null.

Regole per lo svolgimento della prova scritta:

- Per svolgere il compito si hanno a disposizione **100** minuti.
- Scrivere **subito** nome, cognome, matricola su OGNI FOGLIO (**compreso questo**).
- Le risposte al primo esercizio devono essere date direttamente nel riquadro 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
- È possibile consultare appunti, libri e dispense.
- 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 un valido documento di identità.