| Cognome e Nome |
|----------------|
| |
| Matricola |
| |

Programmazione e Algoritmi I A.A. 2018/2019

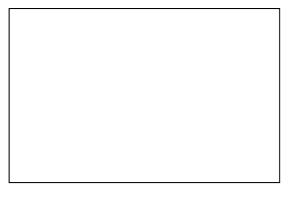
Appello del 2 ottobre 2019

Esercizio 1 (6 punti)

1.1 (3 punti) Cosa stampa il seguente frammento di codice Java?

1.2 (3 punti) Cosa stampa il seguente programma Java?

```
public class Main {
   public static void main(String[] args) {
        System.out.println(enigma(8,2));
        System.out.println(enigma(8,3));
        System.out.println(enigma(1000,4));
   }
   static int enigma (int x, int y) {
        if (x<=0 || y<=0) return 0;
        return 1+enigma(x-y,y);
   }
}</pre>
```



Esercizio 2 (10 punti)

Scrivere un metodo in Java

static void ordinamentoRelativo (int[] a)

che, preso come parametro un array a di numeri interi, ordina gli elementi di a in modo non decrescente dando però precedenza agli elementi pari rispetto a quelli dispari. In altre parole, nell'array ordinato dovranno comparire prima tutti gli elementi pari (ordinati in ordine non decrescente) e poi tutti i dispari (ordinati in ordine non decrescente).

Ad esempio, se **a**={8, 1, 3, 1, 5, 10, 1, 3}, dovrà essere ordinato così: {8, 10, 1, 1, 1, 3, 3, 5}.

Si analizzi la complessità temporale del metodo proposto: tale metodo 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 6 punti totali), dove n è la lunghezza dell'array a in input.

Esercizio 3 (9 punti)

- a. Mostrare **l'heap di minimo** che si ottiene a partire dall'heap vuoto inserendo nell'ordine indicato le chiavi **5, 30, 8, 50, 2, 8, 90, 76, 4** (mostrare l'heap che si ottiene dopo l'inserimento di ogni chiave)
- b. Rimuovere per **3 volte** la chiave minima dall'heap, mostrando l'heap che si ottiene dopo ogni estrazione
- c. Inserire nell'heap così ottenuto le chiavi **35, 27, 6, 88**

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 1)

che, preso come parametro un libretto I, restituisce la media ponderata (rispetto ai CFU) degli esami presenti in I. 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[] estraiStudentiBravi (Libretto[] aL, double soglia) che, preso come parametro un array di libretto aL e un double soglia, sfruttando il metodo mediaPonderata, restituisce un array di interi contenente tutte e sole le matricole degli studenti che abbiano una media ponderata almeno pari a soglia.

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.