

Cognome e Nome \_\_\_\_\_

Matricola \_\_\_\_\_

# Programmazione 1

## A.A. 2015/2016

### Appello del 12 Gennaio 2016

### Compito n° 1

#### Esercizio 1 (9 punti)

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

```
int [] A = {5,41,18,7};
for (int i=A.length-1;i>=0;i--) {
    for (int j=i;j<A.length;j++){
        System.out.print(A[j] + " ");
    }
    System.out.println();
}
```

1.2 (5 punti) Cosa stampa il seguente programma Java?

```
public class Main {
public static void main(String[] args) {
    System.out.println(enigma(1,4));
    System.out.println(enigma(3,-2));
    System.out.println(enigma(6,1000));
}

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

#### Esercizio 2 (9 punti)

- (4 punti) Scrivere un metodo iterativo  
**static boolean isPrime (int n)**  
che, preso come parametro un intero n, restituisce *true* se e solo se n è maggiore di 1 ed è un numero primo.
- (5 punti) Scrivere un metodo iterativo  
**static int[] estraiPrimi (int[] a)**  
che, preso come parametro un array a di numeri interi, restituisce un array di numeri interi (della opportuna lunghezza) contenente tutti e soli gli elementi di a che sono maggiori di 1 e sono numeri primi, rispettando l'ordine e il numero di occorrenze di a.  
Se nessun elemento di a rispetta tali condizioni, viene restituito un array di lunghezza 0.  
Se a vale *null*, viene restituito *null*.  
Ad esempio, se a={1, 4, 3, -7, 5, 6, 3, 2, 8, 4, 4}, viene restituito l'array {3, 5, 3, 2}.

---

### Esercizio 3 (9 punti)

Si consideri il tipo di dato

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

che rappresenta un orario nel formato 24h (le ore variano da 0 a 23, i minuti ed i secondi da 0 a 59).

- **(4 punti)** Scrivere un metodo **static boolean less (Orario o1, Orario o2)** che, prese come parametri due orari **o1** e **o2**, restituisce **true** se e solo se **o1** precede **o2** (assumendo in modo naturale che i suoi orari si riferiscano allo stesso giorno). Se **o1** e/o **o2** valgono *null*, viene restituito **false**.
- **(5 punti)** Scrivere un metodo iterativo **static int conteggio (Orario[] a, Orario o)** che, presa come parametro un array di Orario **a** ed un orario **o** e sfruttando il metodo precedente, restituisce il numero di elementi di **a** che precedono l'orario **o**. Se **a** vale *null*, viene restituito 0. Il metodo deve inoltre gestire in modo opportuno gli elementi dell'array che valgono *null*.

---

### Esercizio 4 (5 punti)

Si consideri il seguente tipo di dato, che rappresenta una coda di numeri interi:

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

```
class Coda{
    Elem first;
    Elem last;
}
```

Scrivere un metodo iterativo

**static int caricoPari (Coda c)**

che, senza modificare la coda, presa come parametro una coda **c** di numeri interi (presenti nel campo carico) restituisce la somma dei carichi di tutti gli elementi di posizione pari (considerando che il primo elemento che sta per uscire dalla coda ha posizione 1, il secondo ha posizione 2 e così via). Se la coda non esiste oppure è vuota, viene restituito **0**.

**Esercizio facoltativo (3 punti):** svolgere l'esercizio 4 in modo ricorsivo, senza far uso di nessun comando iterativo.

#### 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.

Cognome e Nome \_\_\_\_\_

Matricola \_\_\_\_\_

# Programmazione 1

## A.A. 2015/2016

### Appello del 12 Gennaio 2016

### Compito n° 2

#### Esercizio 1 (9 punti)

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

```
int [] A = {10,4,8,3};
for (int i=A.length-1;i>=0;i--) {
    for (int j=i;j<A.length;j++){
        System.out.print(A[j] + " ");
    }
    System.out.println();
}
```

1.2 (5 punti) Cosa stampa il seguente programma Java?

```
public class Main {
public static void main(String[] args) {
    System.out.println(enigma(1,3));
    System.out.println(enigma(2,-5));
    System.out.println(enigma(8,100));
}

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

#### Esercizio 2 (9 punti)

- (4 punti) Scrivere un metodo iterativo  
**static boolean isPrime (int n)**  
che, preso come parametro un intero n, restituisce *true* se e solo se n è maggiore di 1 ed è un numero primo.
- (5 punti) Scrivere un metodo iterativo  
**static int[] estraiPrimi (int[] a)**  
che, preso come parametro un array a di numeri interi, restituisce un array di numeri interi (della opportuna lunghezza) contenente tutti e soli gli elementi di a che sono maggiori di 1 e sono numeri primi, rispettando l'ordine e il numero di occorrenze di a.  
Se nessun elemento di a rispetta tali condizioni, viene restituito un array di lunghezza 0.  
Se a vale *null*, viene restituito *null*.  
Ad esempio, se a={1, 4, 3, -7, 5, 6, 3, 2, 8, 4, 4}, viene restituito l'array {3, 5, 3, 2}.

---

### Esercizio 3 (9 punti)

Si consideri il tipo di dato

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

che rappresenta un orario nel formato 24h (le ore variano da 0 a 23, i minuti ed i secondi da 0 a 59).

- **(4 punti)** Scrivere un metodo **static boolean less (Orario o1, Orario o2)** che, prese come parametri due orari **o1** e **o2**, restituisce **true** se e solo se **o1** precede **o2** (assumendo in modo naturale che i suoi orari si riferiscano allo stesso giorno). Se **o1** e/o **o2** valgono *null*, viene restituito **false**.
- **(5 punti)** Scrivere un metodo iterativo **static int conteggio (Orario[] a, Orario o)** che, presa come parametro un array di Orario **a** ed un orario **o** e sfruttando il metodo precedente, restituisce il numero di elementi di **a** che precedono l'orario **o**. Se **a** vale *null*, viene restituito 0. Il metodo deve inoltre gestire in modo opportuno gli elementi dell'array che valgono *null*.

---

### Esercizio 4 (5 punti)

Si consideri il seguente tipo di dato, che rappresenta una coda di numeri interi:

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

```
class Coda{
    Elem first;
    Elem last;
}
```

Scrivere un metodo iterativo

**static int caricoDispari (Coda c)**

che, senza modificare la coda, presa come parametro una coda **c** di numeri interi (presenti nel campo carico) restituisce la somma dei carichi di tutti gli elementi di posizione dispari (considerando che il primo elemento che sta per uscire dalla coda ha posizione 1, il secondo ha posizione 2 e così via).

Se la coda non esiste oppure è vuota, viene restituito **0**.

**Esercizio facoltativo (3 punti):** svolgere l'esercizio 4 in modo ricorsivo, senza far uso di nessun comando iterativo.

#### 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.