

Nome \_\_\_\_\_

Matricola \_\_\_\_\_

# Programmazione 1

## A.A. 2012/2013

### Appello del 22 Gennaio 2013

#### Compito n° 1

#### Esercizio 1 (9 punti)

**1.1** Cosa stampa il seguente frammento di codice Java?

```
for (int i=0; i<3; i++){
    for (int j=i; j<5; j++){
        System.out.print ((i*j) + " ");
    }
    System.out.println();
}
```

**1.2** Cosa stampa il seguente frammento di codice Java?

```
int [] A = {7,2,5,3,9,1,10,7,12};
int conto = 0, somma = 0;
for (int i=0; i< A.length; i++){
    if (A[i]%2==0){
        somma += A[i];
        conto++;
    }
}
System.out.println((double)somma/conto);
```

**1.3** Cosa stampa il seguente programma Java?

```
public class Main {
public static void main(String[] args) {
    System.out.println(enigma(10));
    System.out.println(enigma(5));
}

public static int enigma (int x){
    if (x<=0) return 0;
    return (2*x-1) + enigma(x-1);
}
}
```

## Esercizio 2 (8 punti)

Dato un array, il suo inverso è dato dall'array in cui la prima cella diviene l'ultima, la seconda la penultima e così via (l'ultima cella diviene la prima).

Dati due array **a** e **b** di numeri interi aventi lo stesso numero di elementi **n**, il prodotto scalare dei due array è dato dalla somma dei prodotti degli elementi corrispondenti.

Ad esempio, il prodotto scalare tra [11, 8, 14] e [2, 5, 1] è  $11*2 + 8*5 + 14*1 = 76$ .

Scrivere un metodo **static int prodottoScalare (int[] a, int[] b)** che presi come parametro due array **a** e **b** di numeri interi restituisce il prodotto scalare di **a** per l'inverso di **b**. Il metodo **non deve** modificare gli array **a** e **b**.

## Esercizio 3 (8 punti)

Scrivere un metodo **static int[][] allocaMatrice (int[] a)** che preso come parametro un array **a** di numeri interi restituisce un nuovo array bidimensionale avente numero di righe uguale alla lunghezza dell'array **a**, e tale che, per ogni  $i=0, \dots, a.length-1$ , la lunghezza della riga di indice **i** sia uguale **i** sia uguale al contenuto della **i**-esima posizione dell'array **a**. Il metodo **non deve** modificare l'array **a**.

## Esercizio 4 (8 punti)

Il **numero magico** di una pila è definito nel seguente modo: finché la pila contiene almeno due elementi, si estraggono i primi due elementi **a** e **b**, e se il primo elemento estratto (**a**) è *pari* si calcola  $c=a+b$ , altrimenti (se è *dispari*) si calcola  $c=a-b$ . L'elemento **c** è quindi inserito nella pila. Quando la pila contiene un solo elemento, questo è il suo numero magico. Quando una pila è vuota, il suo numero magico è **0**.

Date le seguenti classi (viste a lezione) per la rappresentazione in memoria di una pila

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

```
class Pila{
    Elem top;
}
```

e potendo utilizzare (senza riscriverli) i metodi **pop** e **push** per estrarre ed inserire elementi nella pila, scrivere un metodo iterativo **static int numeroMagico (Pila p)** che presa come parametro una pila **p** restituisce il suo numero magico.

**Esercizio facoltativo:** scrivere una versione *ricorsiva* del metodo **numeroMagico** (che non faccia uso di costrutti iterativi)

### Attenzione:

- Per svolgere il compito si hanno a disposizione **90** 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**

Nome \_\_\_\_\_

Matricola \_\_\_\_\_

# Programmazione 1

## A.A. 2012/2013

### Appello del 22 Gennaio 2013

#### Compito n° 2

#### Esercizio 1 (9 punti)

**1.1** Cosa stampa il seguente frammento di codice Java?

```
for (int i=2; i<5; i++){
    for (int j=i; j<8; j++){
        System.out.print ((i+j) + " ");
    }
    System.out.println();
}
```

**1.2** Cosa stampa il seguente frammento di codice Java?

```
int [] A = {11,3,5,4,8,1,10,6};
int conto = 0, somma = 0;
for (int i=0; i< A.length; i++){
    if (A[i]%2==1){
        somma += A[i];
        conto++;
    }
}
System.out.println((double)somma/conto);
```

**1.3** Cosa stampa il seguente programma Java?

```
public class Main {
public static void main(String[] args) {
    System.out.println(enigma(9));
    System.out.println(enigma(6));
}

public static int enigma (int x){
    if (x<=0) return 0;
    return (2*x-1) + enigma(x-1);
}
}
```

## Esercizio 2 (8 punti)

Dati due array **a** e **b** di numeri interi aventi lo stesso numero di elementi **n**, il prodotto scalare dei due array è dato dalla somma dei prodotti degli elementi corrispondenti.

Ad esempio, il prodotto scalare tra [11, 8, 14] e [2, 5, 1] è  $11*2 + 8*5 + 14*1 = 76$ .

Scrivere un metodo **static int prodottoScalare (int[] a, int[] b)** che presi come parametro due array **a** e **b** di numeri interi restituisce il prodotto scalare dei due array. Il metodo **non deve** modificare gli array **a** e **b**.

## Esercizio 3 (8 punti)

Dato un array, il suo inverso è dato dall'array in cui la prima cella diviene l'ultima, la seconda la penultima e così via (l'ultima cella diviene la prima).

Scrivere un metodo **static int[][] allocaMatrice (int[] a)** che preso come parametro un array **a** di numeri interi restituisce un nuovo array bidimensionale avente numero di righe uguale alla lunghezza dell'array **a**, e tale che, per ogni **i=0, ..., a.length-1**, la lunghezza della riga di indice **i** sia uguale al contenuto della *i*-esima posizione dell'array inverso di **a**. Il metodo **non deve** modificare l'array **a**.

## Esercizio 4 (8 punti)

Il **numero magico** di una coda è definito nel seguente modo: finché la coda contiene almeno due elementi, si estraggono i primi due elementi **a** e **b**, e se il primo elemento estratto (**a**) è *pari* si calcola **c=a-b**, altrimenti (se è *dispari*) si calcola **c=a+b**. L'elemento **c** è quindi inserito nella coda. Quando la coda contiene un solo elemento, questo è il suo numero magico. Quando una coda è vuota, il suo numero magico è **0**.

Date le seguenti classi (viste a lezione) per la rappresentazione in memoria di una pila

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

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

e potendo utilizzare (senza riscriverli) i metodi **enqueue** e **dequeue** per inserire ed estrarre elementi nella coda, scrivere un metodo iterativo **static int numeroMagico (Coda c)** che presa come parametro una coda **c** restituisce il suo numero magico.

**Esercizio facoltativo:** scrivere una versione *ricorsiva* del metodo **numeroMagico** (che non faccia uso di costrutti iterativi)

### Attenzione:

- Per svolgere il compito si hanno a disposizione **90** 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**