

**Appello di Programmazione e Algoritmi 1**  
**12 gennaio 2022**

**Esercizio 1**

Cosa stampa il seguente programma Java?

```
public static void main(String[] args) {
    System.out.println(calcola(4));
    System.out.println(calcola(10));
    System.out.println(calcola(2));
}
static int calcola (int x){
    if (x<=0||x>10) return x;
    if (x%2==0) return calcola(x+3) - x;
    return calcola(x+1) + x;
}
```

**NON SARANNO CONSIDERATE VALIDE LE RISPOSTE NON GIUSTIFICATE DAI CALCOLI EFFETTUATI PER OTTENERE IL RISULTATO**

Cosa stampa il seguente frammento di codice Java?

```
boolean flag = true;
int []a = {4,2,-1,-4,6,8};
int num=a.length;
for (int i=1; i<a.length && flag; i++)
{
    if (a[i] <a[i-1])
    {System.out.println("somma: " + (a[i-1]+a[i]));}
    else {flag = false; num=num+a[i-1];}
}
System.out.println("num: " + num);
```

**NON SARANNO CONSIDERATE VALIDE LE RISPOSTE NON GIUSTIFICATE DAI CALCOLI EFFETTUATI PER OTTENERE IL RISULTATO**

**Esercizio 2**

Scrivere un metodo in Java

**static int[] haSuccessore (int[] a, int [] b)**

che preso in input due array **a** e **b** di elementi interi, restituisce un nuovo array in cui sono presenti tutti e soli gli elementi **x** di **a**, tali che **x+1** compare in **b**, mantenendo l'ordine in cui gli elementi compaiono in **a**. Ad esempio: se **a**={5, 7, -9, 5, 8, 5,10, -3, 8} e **b**={6, 3, -8}, il metodo dovrà restituire {5, -9, 5, 5} (il valore 5 di **a** perché in **b** è presente il 6, il valore -9 di **a** perché in **b** è presente -8). Altro esempio, se **a**={1, -2, 3, 1, 5, 2} e **b**={2, 3, -8}, il metodo dovrà restituire {1, 1, 2}. Se **a**={1, -2, 3, 1} e **b**={12, 3, -8}, il metodo dovrà restituire {}.

**Si analizzi la complessità temporale del metodo proposto** (soluzioni con complessità temporale peggiore danno luogo a una valutazione minore).

### Esercizio 3

Scrivere un metodo in Java

**static int[] creaArrayProd (int[] a)**

che preso in input un array **a** di elementi interi, restituisce un nuovo array della stessa lunghezza di **a** in cui l'elemento in posizione *i* è il prodotto degli elementi  $a[i], a[i+1], \dots, a[a.length-1]$  **che hanno valore positivo** (se non ci sono elementi con valore positivo il prodotto è uguale a 1). Ad esempio: se  $a = \{2, -11, 4, 3, -7, 5\}$  il metodo dovrà restituire  $\{120, 60, 60, 15, 5, 5\}$ . Altro esempio, se  $a = \{1, 2, -3, 3, -5, 2\}$  il metodo restituirà  $\{12, 12, 6, 6, 2, 2\}$ .

Se  $a = \{-1, -2, -3, -1, -5, -2\}$  il metodo restituirà  $\{1, 1, 1, 1, 1, 1\}$ .

**Si analizzi la complessità temporale del metodo proposto** (*soluzioni con complessità temporale peggiore danno luogo a una valutazione minore*).

### Esercizio 4

Si consideri il seguente array di numeri interi:

**{12, 6, 10, 13, 8, 24, 18, 11, 4, 22, 17}**

Mostrare **passo-passo l'esecuzione del merge-sort** per ordinare l'array in modo non decrescente.

Si descriva la complessità computazionale del merge-sort (anche scrivendo e risolvendo la ricorrenza  $T(n)$ ).

**[Esercizio di Laboratorio]** SOLO PER GLI STUDENTI CLEII che hanno svolto il progetto dell'AA 2020/2021 In relazione al progetto di laboratorio, si implementino i seguenti due metodi:

- `boolean almenoMeta(boolean[] album)`

che restituisce true se nell'album passato come parametro sono presenti almeno metà delle figurine, false altrimenti;

- `int[] figurineDaScambiare(boolean[] album1, boolean[] album2)`

restituisce un array contenente gli indici di tutte le figurine presenti in album1 e che mancano in album2. Ad esempio, dati in input:

```
album1 = {false, false, true, true, false, true, false, true};
```

```
album2 = {false, true, true, false, false, true, false, false};
```

il metodo restituisce l'array  $\{3, 7\}$ . Si assuma che album1 e album2 abbiano la stessa lunghezza.

