

# Esercizi sull'analisi di complessità degli algoritmi

## (Fondamenti di Informatica 2 – Walter Didimo)

**Esercizio 1** Considera il seguente metodo, che restituisce il massimo valore in un array di interi. Effettuare un'analisi di complessità asintotica del caso peggiore, ed esprimere la complessità con notazione  $O$ , rispetto alla dimensione dell'input.

```
public static int massimo (int[] a){
    int max = a[0];
    for (int i=1; i<a.length; i++)
        if (a[i]>max)
            max=a[i];
    return max;
}
```

**Esercizio 2** Per ciascuno dei seguenti problemi, descrivi un algoritmo e la sua complessità asintotica con notazione  $O$ :

- 1) Data una matrice rettangolare di interi di dimensione  $m \times n$ , visualizzare le somme degli elementi su ogni riga.
- 2) Data una matrice quadrata di dimensione  $n$ , visualizzare la somma degli elementi sulla diagonale principale e quella degli elementi sulla diagonale secondaria.
- 3) Dato un array di  $n$  numeri reali, restituire true se nell'array ci sono due numeri uguali e false in caso contrario.
- 4) Dato un array di  $n$  numeri interi, con valori compresi tra 0 e 100, restituire true se nell'array ci sono due elementi uguali e false in caso contrario.

**Esercizio 3** Analizza la complessità asintotica dei seguenti metodi, in funzione della dimensione dell'input.

```
public static void met1 (int[] a){
    for (int i=0; i<a.length; i++){
        int somma = 0;
        for (int j=i; j<a.length; j++)
            somma += a[j];
        System.out.println (somma);
    }
}
```

```
public static void met2 (char c, String s){
    int f = 0;
    for (int i=0; i<s.length(); i++)
        if (s.charAt(i) == c)
            f++;
    return f;
}
```

```
public static void met3 (int[] a){
    for (int i=0; i<a.length-2; i++){
        int somma = 0;
        for (int j=i; j<i+2; j++)
            somma += a[j];
        System.out.println (somma);
    }
}
```