

## Esempi di Algoritmi in Java

10110

01100

01011

## Verifica di un vettore a valori crescenti

- Scrivere un metodo che verifica se un vettore è a valori reali crescenti. Il metodo ritorna true se la condizione è verificata, altrimenti ritorna false.

```
public static boolean ValoriCrescenti(double[] vett)
{
    boolean risposta= true;
    for (int i = 0; i < vett.length-1; i++)
    {
        if (vett[i+1] <= vett[i])
            { risposta = false; return risposta };
    } // chiude il for
    return risposta;
}
```

10110

01100

01011

## Verifica dell'uguaglianza di somme

---

- Scrivere un metodo che verifica se in un vettore, il cui numero di elementi sia pari, la somma della prima metà degli elementi è uguale alla somma della seconda metà.
- QUINDI:
- Il metodo ritorna **false** se il vettore non è composto da un numero di elementi pari.
- Se il vettore è composto da un numero di elementi pari, il metodo ritorna **true** se la condizione è verificata, altrimenti ritorna **false**.

10110

01100

01011

## Verifica dell'uguaglianza di somme

```
public static boolean VerSemisomme(int [] vett)
{
    int somma1= 0; int somma2=0;
    int mtlung= vett.length/2;
    if (vett.length%2 != 0)
        return false;
    else
    {
        for (int i = 0; i < mtlung; i++)
        {
            somma1 = somma1 +vett[i];
            somma2 = somma2 +vett[mtlung+i];
        }
        if (somma1 == somma2)
            return true;
        else
            return false;
    }
}
```

10110

01100

01011

## Sommatoria di una serie di elementi

- Scrivere un metodo che calcola la sommatoria degli elementi di una serie. Letti  $n$  interi  $i_1, i_2, \dots, i_n$  calcola la seguente somma

$$\sum_{j=1}^n (i_j * i_{(n-j+1)})$$

- Il metodo ritorna il valore della sommatoria.

```
public class Sommatoria
{
    public static int sommatoria(int[] vett)
    {
        int s = 0;
        for (int i = 0; i <= vett.length-1; i++)
            s = s + (vett[i] * vett[(vett.length-1)-i]);
        return s;
    }
    // continua . . .
}
```

10110

01100

01011

# Sommatoria in Java

```
//. . .

public static void main(String args[])
{
    double serie[], somma;
    int l;
    l = Console.readInt("Inserire lung. serie:");
    serie = new double[l];
    for (int i = 0; i<= serie.length-1; i++)
    {
        serie[i]=Console.readDouble("Ins. el. serie: ");
    }
    somma = sommatoria(serie);
    System.out.println(" sommmatoria = " + somma);
}
}
```



10110

## Verifica del prodotto

---

01100

01011

- Scrivere un metodo che verifica se in un vettore, il prodotto degli elementi di posizione pari è maggiore del prodotto degli elementi di posizione dispari.
- La posizione 0 si intende pari.

10110

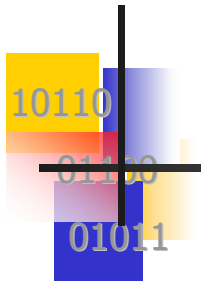
01100

01011

## Verifica del prodotto

```
public static boolean VerProdotto(int [] vett)
{
    int prodpari= 1; int proddispari=1;
    for (int i = 0; i < vett.length; i++)
    {
        if (i%2 == 0)
            prodpari = prodpari * vett[i];
        else
            proddispari = prodispari * vett[i];
    }
    if (prodpari > proddispari)
        return true;
    else
        return false;
}
}
```





## Calcolo di media, massimo e minimo

---

- Scrivere un metodo che
  - dato un vettore  $V$  e
  - un intervallo di valori  $(n, m)$dove  $n$  e  $m$  sono compresi tra 0 e la lunghezza del vettore,  
  
calcola il valore **massimo** e **minimo** e la **media** dei valori del vettore compresi nell'intervallo specificato.

10110

01100

01011

## Calcolo di media, massimo e minimo

```
public static void MaxMinMed (int[] V, int n, int m)
{
    int min, max, somma=0; double med;
    min = V[0];
    max = V[0];
    for (int i = 0; i < V.length; i++)
    {
        somma += V[i];
        if (V[i] < min)
            min = V[i];
        else
            if (V[i] > max)
                max = V[i];
    }
    med = somma/V.length;
    System.out.println("Minimo=  " + min);
    System.out.println("Massimo=  " + max);
    System.out.println("Media=    " + med);
}
```