

Esercizio 3. Scrivere un metodo che, date due stringhe $s1$ ed $s2$, restituisce *true* se la stringa $s1$ è uguale alla stringa $s2$ rovesciata, altrimenti il metodo restituisce *false*. Ad esempio, date $s1 = \text{"orebla"}$ ed $s2 = \text{"albero"}$, il metodo restituisce *true*.

```
public static boolean reverseString (String s1, String s2) {
    if (s1.length() != s2.length())
        return false;
    int n = s1.length();
    for (int i=0; i<n; i++) {
        if (s1.charAt(i) != s2.charAt(n-1-i))
            return false;
    }
    return true;
}
```

Esercizio 4. Scrivere un metodo che, dati un array di interi a (contenente almeno due elementi), un intero v ed un intero k ($k > 0$), restituisce *true* se in a esistono almeno k coppie di interi, la cui somma è uguale a v , altrimenti il metodo restituisce *false*. Ad esempio, se l'array a è $\{5, 2, -4, 7, -2\}$, $v=3$ e $k=2$, il metodo restituisce *true* (in quanto $3=5+(-2)$ e $3=-4+7$).

```
public static boolean checkSum2 (int[] a, int v, int k) {
    if (a.length < 2) return false;
    int cont = 0, i = 0;
    while (i < a.length) {
        int j = i+1;
        while (j < a.length) {
            if (v == a[i]+a[j]) {
                cont++;
                if (cont >= k) return true;
            }
            j++;
        }
        i++;
    }
    return false;
}
```

Esercizio 5. Il *triangolo di Tartaglia* di livello n ($n > 0$) può essere visto come un array bidimensionale di interi definito come segue. Si costruisce il triangolo partendo dal vertice, corrispondente ai livelli 0 ed 1:

```
1
1 1
```

Ogni nuovo livello inizia e termina con 1, mentre i numeri intermedi sono la somma di due interi del livello precedente, come indicato dai segmenti scritti a matita nel seguente triangolo per $n=5$ (si noti che ogni livello k ($0 \leq k \leq n$) è costituito da $k+1$ numeri):

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
```

Scrivere un metodo che, dato un intero n , restituisce un array bidimensionale di interi che rappresenta il triangolo di Tartaglia di livello n .

```
public static int[][] tartaglia (int n) {
    int[][] a = new int[n+1][];

    for (int i=0; i<n+1; i++) {
        a[i] = new int[i+1];
        a[i][0] = 1;
        for (int j=1; j<i; j++) {
            a[i][j] = a[i-1][j-1] + a[i-1][j];
        }
        a[i][i] = 1;
    }
    return a;
}
```

Il ciclo esterno può essere scritto anche nel modo seguente:

```
...
for (int i=0; i<n+1; i++) {
    a[i] = new int[i+1];
    for (int j=0; j<a[i].length; j++) {
        if (j==0 || j==a[i].length-1)
            a[i][j] = 1;
        else
            a[i][j] = a[i-1][j-1] + a[i-1][j];
    }
}
...
```