Thomas Freund
A00831997

```
20 long sumaIterativa( long n){
21     long sum = 0;
22     for(int i=1; i<=n; i++){
23         sum += i;
24     }
25     return sum;
26 }
```

Linea	Comp	Caso
21	1	C1
22	n	C2
22	n	C3
25	1	C4

$$T(n) = C1 * 1 + C2 * n + C3 * n + C4 * 1$$

$$T(n) = (c2 + c3)n + (c1 + c4)\ donde\ a = c2 + c3,\ b = c1 + c4$$

$$T(n) = an + b$$

Dado que se evalúa el peor caso y se lleva al límite donde $\lim_{n \to \infty}$, b se vuelve insignificante.

$$T(n) = an$$

$$Complejidad: O\ (an) = O\ (n)$$

```
28 long sumaDirecta( long n){
29     long sum = 0;
30     sum = ((n * (n + 1)) / 2);
31     return sum;
32 }
```

Linea	Comp	Caso
29	1	C1
30	1	C2

$$T(n) = C1 * 1 + C2 * 1$$

$$T(n) = c1 + c2 = 1$$

$$Complejidad: O\ (1)$$

```
7   long sumaRecursiva( long n){
8    if(n==0){
9      return 0;
10   }
11   return n + sumaRecursiva(n-1);
12  }
```

Linea	Comp	Caso
8	1	C1
9	1	C2
11	1+T(n-1)	C3

Base case:

$$T(n) = 1,\ if\ n = 0$$
$$T(n) = 1 + T(n-1),\ if\ n > 0$$

Since this is a recursive function, we need to find a general solution through a pattern.

$$T(n) = 1 + T(n-1)$$
$$T(n-1) = 1 + 1 + T(n-1-1)$$
$$T(n-2) = 2 + T(n-2)$$
$$T(n-3) = 1 + 2 + T(n-2-1)$$
$$T(n-3) = 3 + T(n-3)$$
$$\ldots$$

General Solution

$$T(n) = k + T(n-k)$$

Using the base case.

$$n - k = 0$$
$$n = k$$

We substitute k:

$$T(n) = n + T(n-n)$$
$$T(n) = n + T(0)$$
$$T(n) = n + 0$$
$$T(n) = n$$

Therefore the complexity of the recursion is:

$$O(n)$$