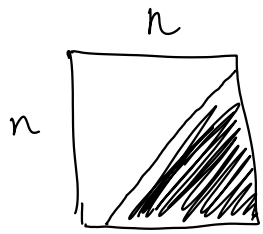
Pseudocode

$O(n^2)$

```
Function isSorted(A, size):
    for each i from 0 to size-2:
        if A[i] > A[i+1]:
            return false
    return true
```

$O(n)$




```
Function isSorted(A, size):
    for each i from 0 to size-1:
        for each j from i+1 to size-1:
            if A[i] > A[j]:
                return false
    return true
```

for each j from 0 to size-1 → for each j from i+1 to size-1

if j > i ...

i	j
0	1, 2, 3, 4 .
1	2, 3, 4 .
2	3, 4 .
3	4 .
4	—

$$\sum_{i=1}^{n} i$$

$$1+2+3+4+\dots+n = \frac{n(n+1)}{2}$$

$$\frac{n(n-1)}{2}$$

n^2

best-case

average case

worst case

space complexity

time complexity

$1+2+3+4+5 = 15$

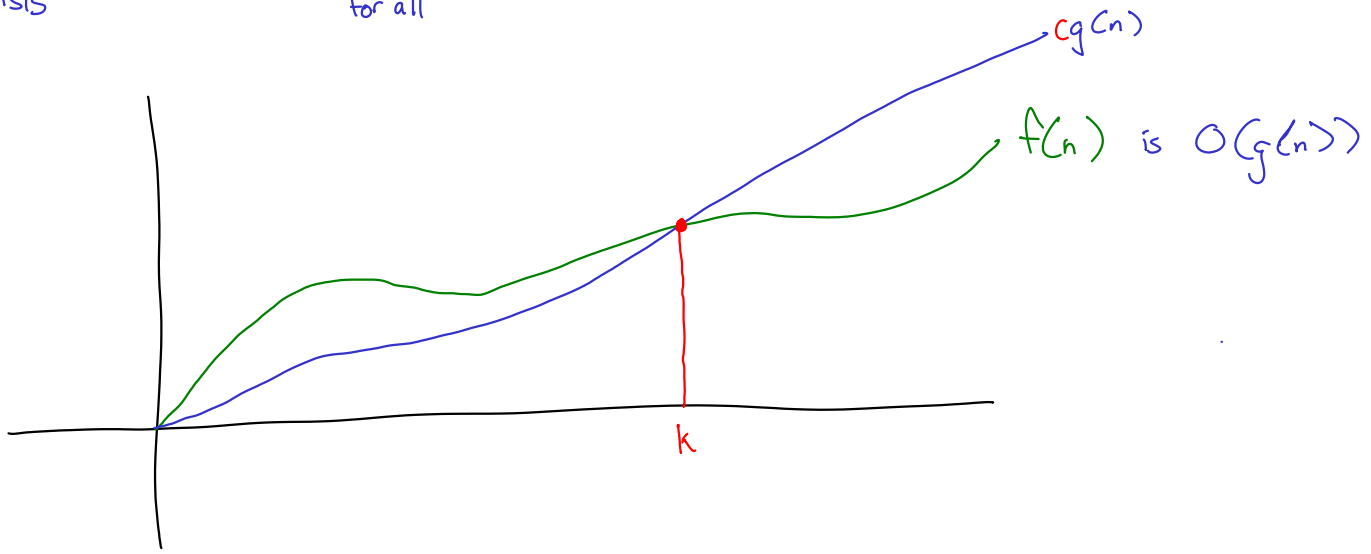
"$n^2 - n + 1$ is $O(n^2)$"

$$\exists\, c \geq 1, k \geq 0.\ \forall n \geq k.\ f(n) \leq cg(n) \equiv f(n) \text{ is } O(g(n))$$

there exists — for all

$\exists\, c \geq 1, k \geq 0.\ \forall\, n \geq k.\ f(n) \leq cg(n) \equiv f(n)$ is $O(g(n))$

there exists — for all



$\exists\, c \geq 1, k \geq 0.\ \forall\, n \geq k.\ f(n) \leq cg(n) \equiv f(n)$ is $O(g(n))$

there exists — for all

$2n^2$ is $O(n^2)$

Choose $c = 3$, $k = 1$: $\forall n \geq 1.\ 2n^2 \leq 3n^2$ ✓

$2n^2 + 1000n + 500$ is $O(n^2)$

$k = 1502$
$c = 2000$

$\forall n \geq 1502.\ 2n^2 + 1000n + 500 \leq 2000n^2$

$\forall n \geq 1502.\ 1000n + 500 \leq 1998n^2$

$\forall n \geq 1502.\ \cancel{1000}n \leq \cancel{1000}n^2$ and $500 \leq 998n^2$

if
$a \leq b$
and
$c \leq d$
then
$a + c \leq b + d$
when all are positive