Keywords:

- dominant order: part of the complexity that cause the greatest effect.

drop the constants. (these have a fixed value)

① ID all variables & variable size.
② ID the operations.
- which are related to the variable size.

Loops:

- count how many times a loop will run
- multiply whatever is nested inside the loop
- Add consecutive loop, multiply nested loops.

Lists:

- list.append(x)	O(1)	constant
- list[i] = x	O(1)	constant
- x in list	O(n)	linear

time complexity

→ space = len(list) x size of item in list.

Dicts: dict[key] = value O(1)] time
key in dict O(1)

space = (# of keys·values) * size of the key) + ((# of values) * size of value)

Str

str = "something"	O(# of chars)
str += str2	O(# of char + # of chars in str2)

time

- space O(# of char)