

Big Oh

Intro/Motivation

Efficiency

Random access — means we can access any elt in an array/list immediately.

What is efficiency?

★ Time efficiency: measure how much time the alg takes to run.

Space efficiency: " " " memory the alg uses

(Example 1)

Count "basic operations"

```
float mean(int a[], int n) {
```

① sum = 0;

(line 1)

① i = 0;

(line 2)

① while (i < n) {

(line 3)

① sum += a[i];

(line 4)

① i++;

(line 5)

}

① return sum / n;

(line 6)

} $3n + 4$

Def'n of big-oh

→ formula in terms of basic operations

$T(n)$ is $O(f(n))$ if and only if

$\underbrace{\hspace{1cm}}$ big-oh time

$$3n+4 = O(n)$$

there exists some constant c such that

$T(n) \leq c * f(n)$ for all sufficiently large values of n .

Even more formally:

$T(n)$ is $O(f(n))$ if and only if

$$\exists c, n_0 \forall n > n_0 T(n) \leq c * f(n)$$

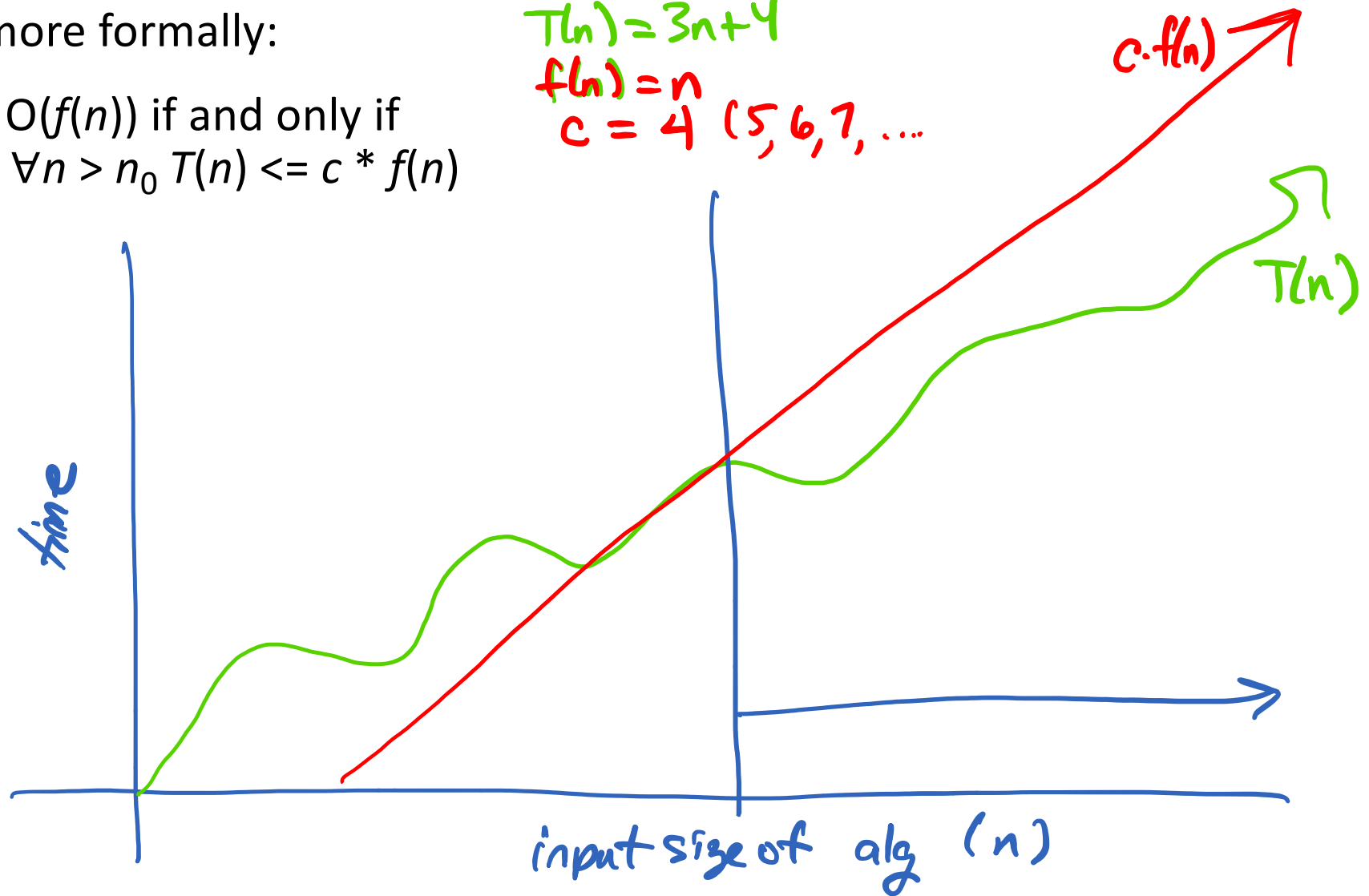
$T(n)$ is $O(f(n))$ if and only if

there exists some constant c such that
 $T(n)$ $\leq c * f(n)$ for all sufficiently large values of n .

Even more formally:

$T(n)$ is $O(f(n))$ if and only if
 $\exists c, n_0 \forall n > n_0 T(n) \leq c * f(n)$

$$\begin{aligned} T(n) &= 3n + 4 \\ f(n) &= n \\ c &= 4 \quad (5, 6, 7, \dots) \end{aligned}$$



~~Let's show $T(n) = 3n+4 = O(n)$~~

$$T(n) = 3n^2 + 4n + 2$$

$$T(n) = O(n) \quad \times \text{ FALSE}$$

$$T(n) \not\leq c \cdot n$$

True $3n+4 = O(2n)?$

True ~~False~~ $3n+4 = O(n^2)?$

~~False~~ $3n+4 = O(\sqrt{n})?$