- Reminder about online sources (like Chegg.com)...
- PS3 extension — 24 hours!

Example 1:

```
def is_prime(n: int) -> bool:
    # Precondition: n >= 2
    for d in range(2, n):
        if n % d == 0:
            return False
    return True
```

\# iterations?

1 (constant)

1

Runtime?

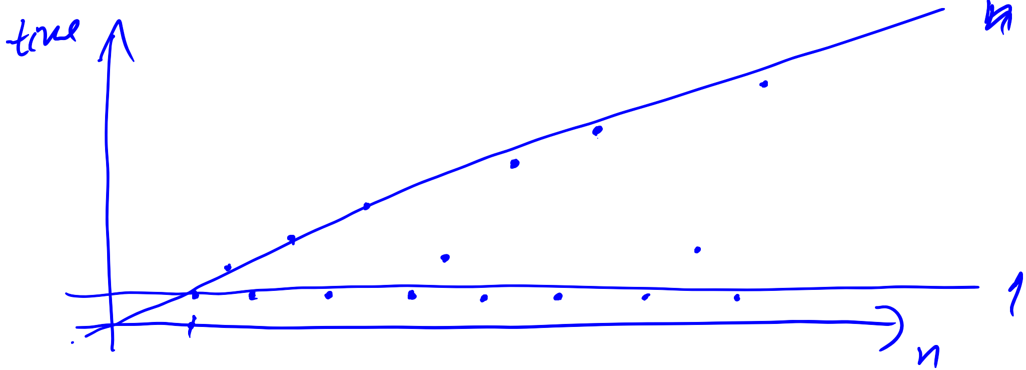
Here input = "size" so there is only <u>one</u> case for each n

loop performs between 1 and $n-2$ iterations
— depends on value of n . . .

Conclusion:

- runtime is in $O(n)$:
  \# iterations $\leq n-2$ and everything else is constant

- runtime is in $\Omega(1)$:
  \# iterations could be as small as 1



NO
Θ expression
for runtime...

Example 2:

```python
def print_primes(n: int) -> None:
    for k in range(2, n+1):
        if is_prime(k):
            print(k)
```

Runtime?

Instead of trying to find a Θ-expression directly, split this up into an upper bound (O) and a lower bound (Ω)

Note: at no point do we try to come up with an "exact" expression for runtime upper bound (O): overestimate

- \# iterations $= n-1 \le n$
- runtime for each iteration:
  depends on runtime of is_prime(k)

In general, when calling a function,
(1) figure out the input size for function call
(2) count runtime for that function as part of your runtime

Here, is_prime(k) runs in time $O(k)$
so one iteration takes time $\le c \cdot k \le c \cdot n$
$(k \le n)$
Total $\le c \cdot n^2 \Rightarrow O(n^2)$

Lower bound (Ω): underestimate

- # iterations $= n-1$
- time for each iteration is ≥ 1
- total is $\geq n-1 \Rightarrow \Omega(n)$

When faced with such a gap, try to do a more careful analysis...

Notation:

$RT_{ip}(n)$ = runtime of is_prime on input n

$RT_{pp}(n)$ = runtime of print_primes on input n

Upper bound:

$$RT_{pp}(n) = \sum_{k=2}^{n} \left(RT_{is}(k) + 1 \right)$$

extra work for loop and print

$$\leq \sum_{k=2}^{n} k + 1 \qquad \left(RT_{is}(k) \in O(k) \right)$$

$$\leq \frac{(n+1)(n+2)}{2} \in \underline{O(n^2)}$$

Lower bound:

$$RT_{pp}(n) = \sum_{k=2}^{n} \left(RT_{is}(k) + 1 \right)$$

$$= \sum_{\substack{2 \leq k \leq n \\ k \text{ is prime}}} \left(RT_{is}(k) + 1 \right) + \cancel{\sum_{\substack{2 \leq k \leq n \\ k \ \underline{\text{not}} \text{ prime}}} \left(RT_{is}(k) + 1 \right)}$$

$$\geq \sum_{\substack{2 \leq k \leq n \\ k \text{ is prime}}} (k+1) \geq \boxed{\sum_{\substack{2 \leq k \leq n \\ k \text{ is prime}}} k}$$

NEED external fact $\approx \dfrac{n^2}{\log n}$

(because there are $\approx \dfrac{n}{\log n}$ primes $\leq n$)

$\Rightarrow$ total is in $\underline{\Omega\left(\dfrac{n^2}{\log n}\right)}$

Still no $\Theta\left(O(n^2) - \Omega\left(\dfrac{n^2}{\log n}\right)\right)$.