- PS2? Will be marked by the weekend!
- TT2? Hard to say... the TAs are working hard but it's still too early to tell...
- PS1 & TT1 remarks: same as TT2... Updates will be provided on Piazza

---

Algorithm analysis — runtime

Goal: Given an algorithm, find a simple function $f : \mathbb{N} \to \mathbb{R}^{\geq 0}$ such that the "running time" of the algorithm is in $\Theta(f)$

— where "running time" = # steps executed, as a function of input size

- Input size?

- rigorously, "standard size" = total # of bits required to represent the entire input
- in practice, we rely on conventions
  1. integers have fixed size
     (really that's not true)
  2. strings have size = # characters
  3. lists have size = length
     (again, "true size" = sum of the sizes of all elements)

- # steps?

- 1 step = <u>any</u> group of statements that always execute together and whose runtime <u>does not depend on input size</u>

NOTE: THERE IS <u>NO</u> ABSOLUTE DEFINITION OF "ONE STEP"

→ what is NOT constant?

- loops
- function calls

} CSC165

- recursion — CSC236
- complex data structures — CSC263, a little in CSC148

Example:

```
def f(n: int) -> int:     # Assume n ≥ 0
    r = 0
    for i in range(10):           # Loop 1
        for j in range(n * n):    # Loop 2
            r = r + j
    for i in range(n // 2):       # Loop 3
        for j in range(i * i):    # Loop 4
            r = r - j
    return r
```

NOTE: In CSC165, when we analyse

runtime of algorithms with a single integer input — express the runtime as a function of the <u>value</u> of the input, not its size.

- analyze loops <u>inside-out</u>

- loop 2: (lines 3–4)
  - body takes time: 1 (line 4)
  - loop iterates n^2 times
  - total time is $\underbrace{1+1+\dots+1}_{n^2} = n^2$

- loop 1: (lines 2–4)
  - body (lines 3–4) takes time $= n^2$
  - # iterations = 10
  - total time $= \underbrace{n^2+n^2+\dots+n^2}_{10} = 10n^2$

- loop 4 (lines 6–7): – body takes time: 1 (line 7)
  – # iterations: i^2
  – total time: $1 \cdot i^2 = i^2$

- Loop 3 (lines 5–7): – body (lines 6–7) takes time: i^2
  – # iterations: $\left\lfloor \frac{n}{2} \right\rfloor$
  – total time = $\underbrace{0^2 + 1^2 + 2^2 + \ldots + \left(\left\lfloor \frac{n}{2} \right\rfloor - 1\right)^2}_{\text{all the values of } i \text{ in range}(n//2)}$

$$= \sum_{i=0}^{\lfloor \frac{n}{2} \rfloor - 1} i^2 = \frac{\left(\lfloor \frac{n}{2} \rfloor - 1\right)\lfloor \frac{n}{2} \rfloor \left(2\lfloor \frac{n}{2} \rfloor - 1\right)}{6}$$

$$= \ldots = \underbrace{a n^3 + b n^2 + c n + d}$$

$$\text{Overall: } \underbrace{\underset{\text{(lines 2-4)}}{10n^2} + \underset{\text{(lines 5-7)}}{an^3+bn^2+cn+d} + \underset{\text{(lines 1, 8)}}{1}}_{\Theta(n^3)}$$