CS61A

MUTABILITY, ITERATORS, AND GENERATORS

LOGISTICS AND REMINDERS

- Lab06 due **Today**
- HW04 Due **Tomorrow** (Please complete the survey)
- Midterm 2 is 10/27 ~3 weeks
- Lab Stuff(Hw, Projs, etc.)

AGENDA

- Mutability
- Iterators
- Generators

ITERABLES AND ITERATORS

- **Iterable**: Something you can call **iter** on
  - ex. lists because **iter([1,2,3])** valid
- **Iterators**: Something you can call **next** on
  - ex. values returned from iter such as **next(iter([1,2,3])**
- Are these mutually exclusive?
- StopIterationError

GENERATORS

- Special kinds of iterators
  - Iterators you can customize!
- Defining trait:
  - function that contains **yield** or **yield from**
- Good Exam Problem: Spring ‘18 Final Q4

```
def naturals():
    i = 0
    while True:
        yield i
        i += 1
```

```
def naturals(n):
    yield n
    yield from naturals(n+1)
```

CONSEQUENCES OF GENERATORS

- A notion of infinite sequences
- Like with naturals you can call **next** as many times as you want
- Naturals never ends. Try making integers or rationals

TEXT

WHY DO WE CARE ABOUT ITERATORS?

- What if we have a list of 10 millions elements or more?
- What kinds of situations do things like this arise?
  - Machine Learning, Big Data
    - Sentences may be encoded as lists of numbers, now imagine how many words are on wikipedia
    - Images can also be encoded as lists. In biomedical imaging images can be 10,000 x 10,000 pixels
      - 100,000,000 pixels around -> 100,000,000 bytes -> 800MB ~ 1GB
      - Most of your computers probably have around 8GB of ram so RIP Laptops
      - A training set of 1000 images would be ~1TB in size