

## Lists and Tuples and Sets (Oh My)

- **Lists are versatile, useful, wonderful, mutable**
  - Given list, cannot stop client code/user from changing it
  - Sometimes mutable data is NOT a good thing
    - Strings are immutable, can be very efficient at large scales
    - Lists are mutable, cannot offer same efficiencies
- **Tuples are similar to lists, but immutable**
  - Cannot add a list to a set, can add a tuple to a set
  - We will see later: very, very useful in dictionaries
  - Useful in other contexts as well: immutable (poker code)
- **See code examples**

Compsci 06/101, Spring 2011

9.1

## From Lists to Tuples

- lists, lints, lines, fines, fixes, fixed
- lists, fists, fiats, feats, seats, stats, stars, stark, stack, stuck

```
def get_roll():  
    return [random.randint(1,6), random.randint(1,6)]
```

```
def track(n):  
    store = set()  
    for x in range(n):  
        store.add(get_roll())  
    return len(store)
```

```
store.add(get_roll())  
TypeError: unhashable type: 'list'
```

Compsci 06/101, Spring 2011

9.2

## Tuomas Sandholm, CMU Professor

- IJCAI Computers and Thought 2003  
Organ Network uses Carnegie Mellon algorithm to match live kidney donors with recipients National Pilot Program facilitates kidney paired-donation transplants



- <http://bit.ly/9No5S5>  
Using game theory, a group of computer scientists has developed a set of algorithms to help thwart terrorist attacks by randomizing where and when security checkpoints, officers, canine units and other deterrents are located in and around the airport.

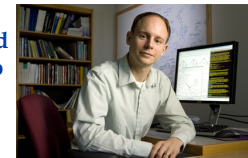
Compsci 06/101, Spring 2011

9.3

## Vince Conitzer @ cs.duke.edu

Vincent Conitzer will receive the 2011 Computers and Thought Award ... The award is presented every two years to the world's leading AI researchers under the age of 35.

Conitzer, an assistant professor of computer science at Duke University, is receiving the award in recognition of his seminal work at the boundary of microeconomic theory and artificial intelligence, in particular for groundbreaking work on computational aspects of game theory, social choice, and mechanism design.



Compsci 06/101, Spring 2011

9.4