Lambdas, Environments, Midterm Review

David E. Culler
CS8 – Computational Structures in Data Science
http://inst.eecs.berkeley.edu/~cs88

Administrative Issues

• Midterm exam: wed Oct 3 6-8 pm
  – Room based on last digit of SID
  – 6-5 LeConte 1 (60%)
  – 5-9: VLSB 2040
  – Alternative and accommodations during 5-9 by request
• Materials will go through 10/1 Lecture
• Please do mid-term survey
• Office hours start here after class and migrate down to BIDS in 190 Doe Library
• Live piazza thread 166

Computational Concepts Toolbox

• Data type: values, literals, operations.
  – e.g., int, float, string
• Expressions, Call expression
• Variables
• Assignment Statement
• Sequences: tuple, list indexing
• Data structures
• Tuple assignment
• Call Expressions
• Function Definition Statement
• Conditional Statement
• Iteration:
  – data-driven (list comprehension)
  – control-driven (for statement)
• Higher Order Functions
  – Functions as Values
  – Functions with functions as argument
  – Assignment of function values
• Recursion
• Lambda - function valued expressions

Recall Tree Recursion with HOF

```python
def qsort(s):
    """Sort a sequence - split it by the first element, sort both parts and put them back together."""
    if not s:
        return []
    else:
        pivot = first(s)
        lessor, more = split_fun(leq_maker(pivot), rest(s))
        return qsort(lessor) + [pivot] + qsort(more)
```

```python
>>> qsort([3,3,1,4,5,4,3,2,1,17])
[1, 1, 2, 3, 3, 4, 4, 5, 17]
```

Exploring Environments

```python
lambda <arg or arg_tuple> : <expression using args>
```

```python
inc = lambda v : v + 1
def incr(v):
    return v + 1
```
Lambda Examples

```python
>>> sort([1,2,3,4,5], lambda x: x)
[1, 2, 3, 4, 5]
```

```python
>>> sort([1,2,3,4,5], lambda x: -x)
[5, 4, 3, 2, 1]
```

```python
>>> sort(((2, 'hi'), (1, 'how'), (5, 'goes'), (7, 'I')), lambda x: x[0])
[(7, 'I'), (5, 'goes'), (2, 'hi'), (1, 'how')]
```

```python
>>> sort(((2,'hi'),(1,'how'),(5,'goes'),(7,'I')), lambda x: len(x[1]))
[(7, 'I'), (5, 'goes'), (2, 'hi'), (1, 'how')]
```

http://cs88-website.github.io/assets/slides/adt/mersort.py

Lambdas

```python
>>> def inc_maker(i):
...     return lambda x: x+i
...>>> inc_maker(3)
<function inc_maker.<locals>.<lambda> at 0x10073c510>
```

```python
>>> inc_maker(3)(4)
7
```

```python
>>> map(lambda x: x*x, [1,2,3,4])
<map object at 0x1020950b8>
```

```python
>>> list(map(lambda x: x*x, [1,2,3,4]))
[1, 4, 9, 16]
```

Thinking back over concepts

- **Data type**
  - Representation
    - Literal and display
    - Internal representation
  - Set of operations
  - Conversions to other types
- **Expressions** – computation of values of a type
  - Built-in operations and function calls
  - Comprehensions
- **Statements**
  - Assignment & Control
  - Conditionals, iteration
- **Functions** – objects and control