Project 4 Contest Gallery

Prizes will be awarded for the winning entry in each of the following categories.

- **Featherweight.** At most 128 words of Logo, not including comments and delimiters.
- **Heavyweight.** At most 1024 words of Logo, not including comments and delimiters.

Winners will be selected by popular vote! (Homework 13)

- Static images of the output of your programs
- Tonight at midnight: I'll post your Logo implementations!
  - Run them to see these images evolve!
- I will also post a solution to the Logo project
  - It runs (almost) all of the contest entries
  - You can use it as a study guide for the final

MapReduce

MapReduce is a framework for batch processing of Big Data

What does that mean?
- **Framework:** A system used by programmers to build applications
- **Batch processing:** All the data is available at the outset and results aren't consumed until processing completes
- **Big Data:** A buzzword used to describe datasets so large that they reveal facts about the world via statistical analysis

The big ideas that underly MapReduce:
- Datasets are too big to be stored or analyzed on one machine
- When using multiple machines, systems issues abound
- Pure functions enable an abstraction barrier between data processing logic and distributed system administration

The Unix Operating System

Essential features of the Unix operating system (and variants)
- **Portability:** The same operating system on different hardware
- **Multi-Tasking:** Many processes run concurrently on a machine
- **Plain Text:** Data is stored and shared in text format
- **Modularity:** Small tools are composed flexibly via pipes

The standard streams in a Unix-like operating system are conceptually similar to Python iterators

Python Programs in a Unix Environment

The built-in `input` function reads a line from standard input.

The built-in `print` function writes a line to standard output.

The values `sys.stdin` and `sys.stdout` also provide access to the Unix standard streams as "files."

A Python "file" is an interface that supports iteration, read, and write methods.

Using these "files" takes advantage of the operating system standard stream abstraction.
MapReduce Evaluation Model

Map phase: Apply a mapper function to inputs, emitting a set of intermediate key-value pairs
- The mapper takes an iterator over inputs, such as text lines.
- The mapper yields 0 or more key-value pairs per input.

Reduce phase: For each intermediate key, apply a reducer function to accumulate all values associated with that key
- The reducer takes an iterator over key-value pairs.
- All pairs with a given key are consecutive.
- The reducer yields 0 or more values for a key, each associated with that intermediate key.

Above-the-Line: Execution model

Below-the-Line: Parallel Execution

Google MapReduce
Is a Big Data framework
For batch processing

Python Examples of a MapReduce Application

The mapper and reducer are both self-contained Python programs
- Read from standard input and write to standard output!

Mapper

```python
#!/usr/bin/env python3
import sys
from ucb import main
from mr import emit

def emit_vowels(line):
    for vowel in 'aeiou':
        count = line.count(vowel)
        if count > 0:
            emit(vowel, count)

@main
def run():
    for line in sys.stdin:
        emit_vowels(line)
```

Reducer

```python
#!/usr/bin/env python3
import sys
from ucb import main
from mr import emit, values_by_key

@main
def run():
    for key, value_iterator in values_by_key(sys.stdin):
        emit(key, sum(value_iterator))
```

Python Examples of a MapReduce Application

The mapper and reducer are both self-contained Python programs
- Read from standard input and write to standard output!
What Does the MapReduce Framework Provide

**Fault tolerance**: A machine or hard drive might crash
- The MapReduce framework automatically re-runs failed tasks.

**Speed**: Some machine might be slow because it's overloaded
- The framework can run multiple copies of a task and keep the result of the one that finishes first.

**Network locality**: Data transfer is expensive
- The framework tries to schedule map tasks on the machines that hold the data to be processed.

**Monitoring**: Will my job finish before dinner?!?
- The framework provides a web-based interface describing jobs.