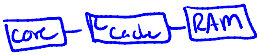
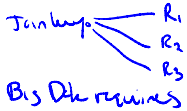
Midterm

- Computer Architecture
  - CPU is fastest, faster than memory



- Joins
  - Joins are expensive - quadratic input comparisons
  - SQL database optimizes join sequence for us
    - Filters first then join
- Maps
  - Maps/dictionaries: {key: value, key2: value2}
  - Tuples (unchangeable) can have keys
  - Map is much faster than |S| + |T|
  - Joining more than one field e.g. (first + last name)
  - Cartesian n×m



  - Big Data requires reading 1 block at a time
- Making dataframes more efficient:
  - Order of evaluation
  - minimize intermediate results
  - SQL database does the best sequence for user
  - Pandas we need to control it on our own
- * Joins iterate over both tables
  - Indexes makes lookup more efficient if join is on index key
- Large volumes of data
  1. Reduce amount of data
  2. Use more efficient algos or data structures
  3. Split work among multiple processors
     - Modern computers have 4 CPU cores
- Parallelism - Multicore Processing
  - apply function on column uses parallelism by by running function on different rows on different cores
  - results then merged together
  - DASK parallel PANDAS
    - partitions data
    - partition (e.g. 100) then merge
    - partitions needs to be more than # of cores
  - Google Colab runs on multicore machine
  - Pandas in itself doesn't use multicore
- Relational Algebra
  - Groupby
  - join
  - select
  - apply
  - merge