\[ M \cdot x = x \cdot y \]
\[ \text{if } \lambda \text{, } M \text{ is } n \times n \text{ matrix } M \text{ vector} \]

given \( M \) and a vector \( v \)
compute \( y = M \cdot v \)
\[ y_i = \sum_{j=1}^{n} M_{ij} \cdot v_j \]

Assumption:
\( y \) could fit in memory but \( M \)
cannot fit in memory if \( M \) is sparse.

If \( M \) is sparse, you can represent the elements of \( M \) as \((i, j, m_{ij})\)

Map: Input \((i, j, m_{ij})\)
output: \((i, \sum_{j} m_{ij})\)

Reduce: \( \sum_{\text{element } x \in \ell} x \)
for each \text{element } x \in \ell
\sum += x
return \((i, \sum)\)
Eigenvectors and eigenvalue:

\( R(A_1, \ldots, A_n) \)

Relational Algebra defines some ops.
on relational tables. (i.e. R)
\( \sigma_C(R) \rightarrow \text{select tuples that satisfy } C. \)
\( \Pi_A(R) \rightarrow \text{projection } \mid \text{RUS, LNS, R-S} \)

\( R_1 \times R_2 \rightarrow \text{cartesian product} \)
\( R_1 \bowtie R_2 \rightarrow \text{natural join of two table} \)
\[ R_1 \bowtie R_2 = \sigma_C(R_1 \times R_2) \]
\( C \) check all common attributes are equal.

\( \phi(R) \)
\( \phi(\text{student table}) \)
Nationality, count(A)

where \( x \) is a grouping attribute

or \( f(A) \) where \( f \in \{ \text{sum, count, min, max, avg, by} \} \)
$\mathcal{C}(L)$ on map-reduce

map: input is a tuple $t$ of $L$
output: if $t$ satisfies $C$ (condition) than output $(t_1, t)$
reduce: $(t_1, t)$
does nothing

$\mathcal{T}_A(L)$ on map-reduce

map: input tuple $t$
project attributes of $t$ based on $A$.
Let's call projected values $t'$
output: $(t', t')$
key value
reduce $(t_1$ as key and list of $(t_1, ... t')$
reduce will convert 
\[(t_1, [t_1, \ldots, t_1])\] to 
\[(t_1, t_1)\]

output \((t_1, t_1)\)

union : \(R \times S\)

map: output \((t, t)\) for both \(R \times S\).

reduce : \((t_1, [t_1, \ldots, t_1])\)
output one \((t_1, t)\) \(\rightarrow\) union

For intersection: change the reduce 
such that if the value list 
has two tuples than output 
\((t, t)\)

Assume we want to compute \(L - S\)
Map: output \((+, R)\) for \(R\)
output \((+, S)\) for \(S\).

Reduce: input \((+, l)\)

If \(l = [E, S] \mid l = [S_i, R]\)
output nothing

else \(l = [S]\)
output nothing

else (if \(l = [E]\))
output \((+, t)\)

\[\text{RMS: } 2CA, 1B) \quad SCB, C)\]

Map: output \((b, (a, R))\) for \(R\)
\((b, (c, S))\) for \(S\)

Reduce: input \((b, l)\)
for each \((c_i, 2)\) in \(l\)
for each \((g_i, S)\) in \(l\)
create \((a_i, b, c_i)\)
Output: \[
(b, (a_1, b, c_1), (a_2, b, c_2), (a, b, c))
\]

Grouping & Aggregation by map-reduce

given \( L(\alpha, \beta, \gamma) \) \( \Gamma(\eta, \Phi(\epsilon)) \)

\( L(\text{count, avg, } \ldots) \)

Map: for each tuple \( t \) of \( L \)

output \( (a, b) \)

Reduce: input \( \rightarrow [a, \sum b_i, \ldots, b_k] \)

output \( \rightarrow (a, \Phi(b_1, \ldots, b_k)) \)

Given \( M \) is \( n \times t \) matrix

\( N \) is \( t \times r \) matrix

Our goal is to compute \( P = M \cdot N \)
\[
P_{ik} = \sum_{\tau=1}^{+} (m_{\tau i} \cdot \eta_{5k})
\]

MC \(I, f, u\) \(N\) \(j, k, w\)

for each \((r, f, m_{rf})\)

output \((f, (M, i, m_{rf})\)

for each \((f, k_i, m_{fk})\)

output \((f, (N, k_i, m_{fk})\)

reduce:
Input \((f, l)\)

output \((f, (i, k_l, m_{if} \cdot \eta_{5k_l})\)

second map-reduce step
map: \((key, value)\)

Output \((i, k_l, v)\) for each \((i, k_l, v)\) in the value list.
reduce
input: (i; k), [u_{1}, \ldots, u_{t}]
output: (i; k), \frac{\sum u_{i}}{n_{i}}

Exercise: Do this only with one map-reduce!