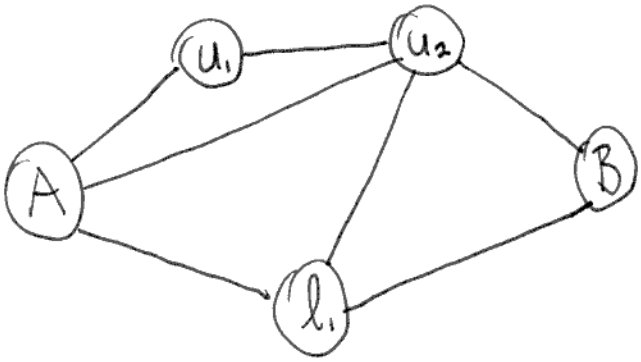
Intended Solution

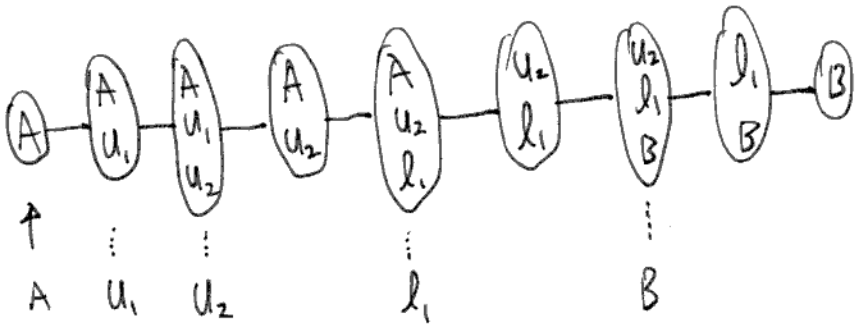
- iterate over bags X_i
- at each "introduce" bag, compute $d(x, v)$
  for v the new node and x each node in X_i, current bag
- then compute, for each $y \in G$ already visited:

$$d(y, v) = \min_{x \in X_i} \left(d(y, x) + d(x, v) \right)$$

$O(k \cdot n)$ per bag.



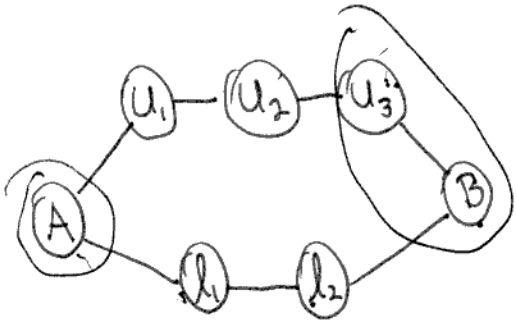
Nice Path Decomposition:



New node: A u_1 u_2 l_1 B

A path paved with good intentions

When does this break?



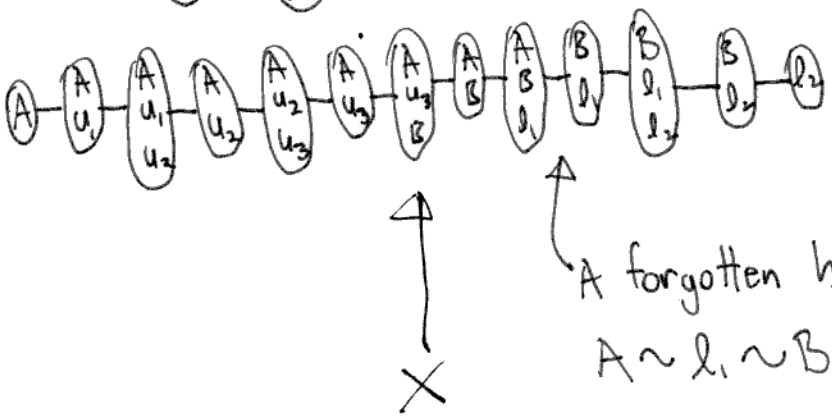
distances computed

$A, u_1 = 1$ ✓

$A, u_2 = 2$ ✓

$\vdots$

$A, B = 4$ ✗



A forgotten here, we never check

$A \sim l_1 \sim B$ = length 3 path.

✗

If it's broke, fix it

Lessons:

- research is hard and involves a lot of trial and error
- when something doesn't work, try fixing by checking your assumptions!
  - did I assume something I didn't realize?
  - can we introduce an assumption that makes things work?