ICPC模擬国内予選2017
F: マトリョーシカ

原案: darsein

問題文: torus

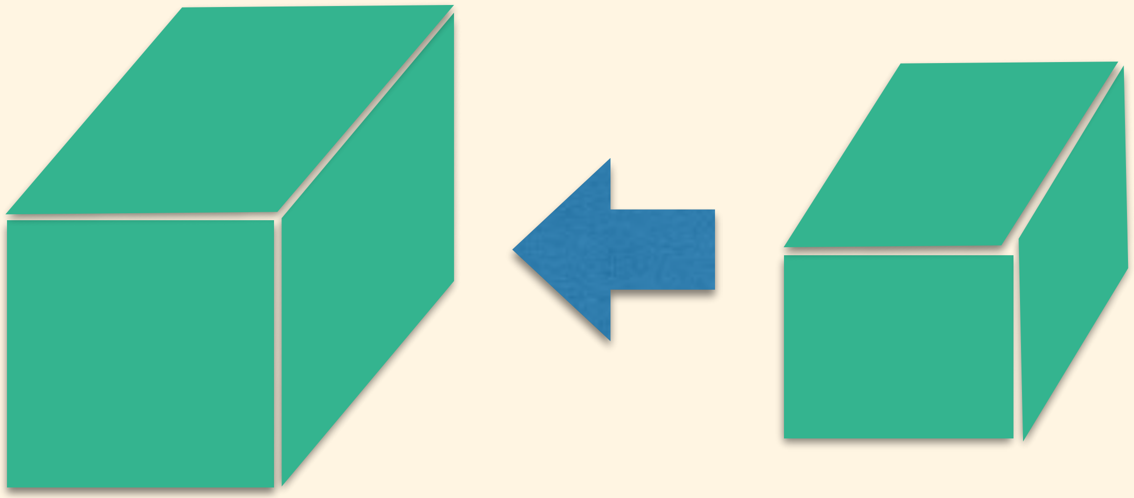
解答: darsein, not, tokoharu, torus

解説: darsein

問題概要

- N体のマトリョーシカがあり、それぞれ$x_i \times y_i \times z_i$ の直方体である
- マトリョーシカは、他のマトリョーシカの中に収納できる
  - 1体の中に直接収納できるのは1体のみ
  - 各辺がそれぞれ平行 (x, y, z は回転して変えてもよい)
  - 収納される方が各辺の長さが真に短い
- 最終的に収納されないマトリョーシカの体積和を最小化せよ

- 制約: $1 \le N \le 100$, $1 \le x_i, y_i, z_i \le 100$



収納できるか否かの判定

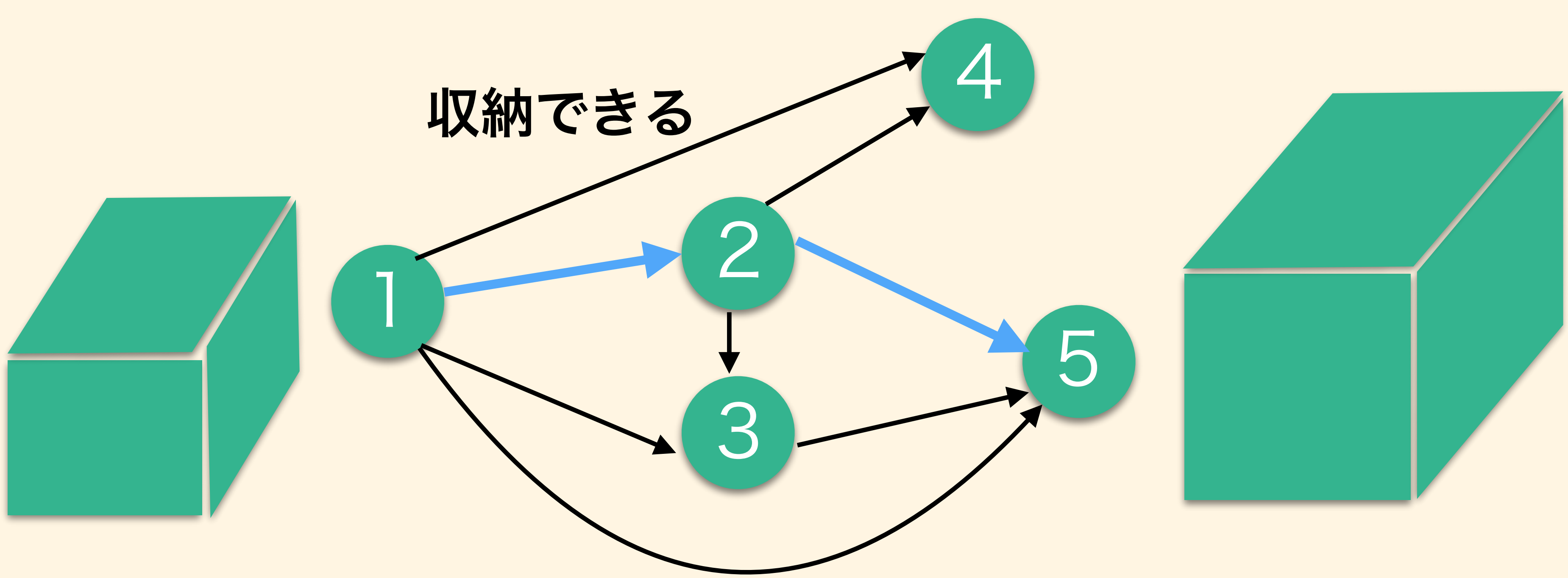
- 長方形Aを長方形Bに収納できる ⇔
  それぞれの3辺を昇順に比較していってどれもAの辺の方が短い

  - →) Aの3辺が$a_1 \leq a_2 \leq a_3$、Bの3辺が$b_1 \leq b_2 \leq b_3$のとき、例えば収納できる辺の合わせ方が $(a_1, a_2, a_3) \Leftrightarrow (b_1, b_3, b_2)$ だとすると、$a_2 \leq a_3 < b_2$、$a_3 < b_2 \leq b_3$よりb_2とb_3を入れ替えてもOK。これをやりまくれば昇順になるのでOK。

  - ←) そう合うように回転すれば入るので自明

単純な場合: 見える個数を最小化

- 体積は無視で個数を最小化する問題を考える
- マトリョーシカを頂点とし、収納できるか否かの関係を辺で表すグラフを構築する
- グラフ上の1つのパス ⇔ 1つのマトリョーシカの中に (再帰的に) 収納できる集合



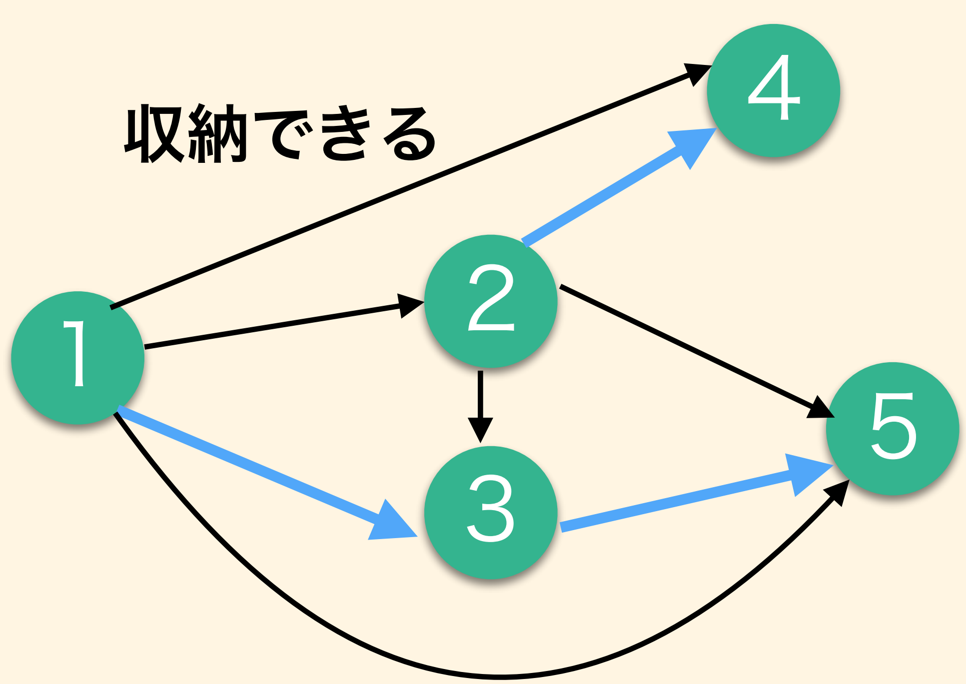
単純な場合: 見える個数を最小化

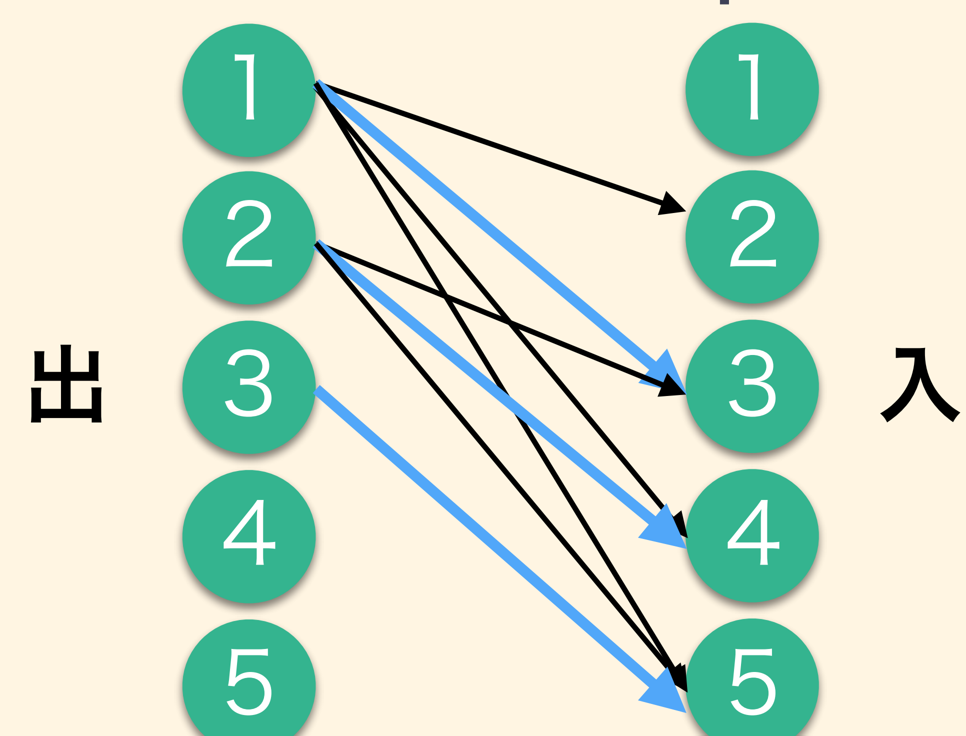
- 被覆するパスの個数が最小

  ⇔ 見えているマトリョーシカの個数が最小

  → DAGの最小パス被覆問題！

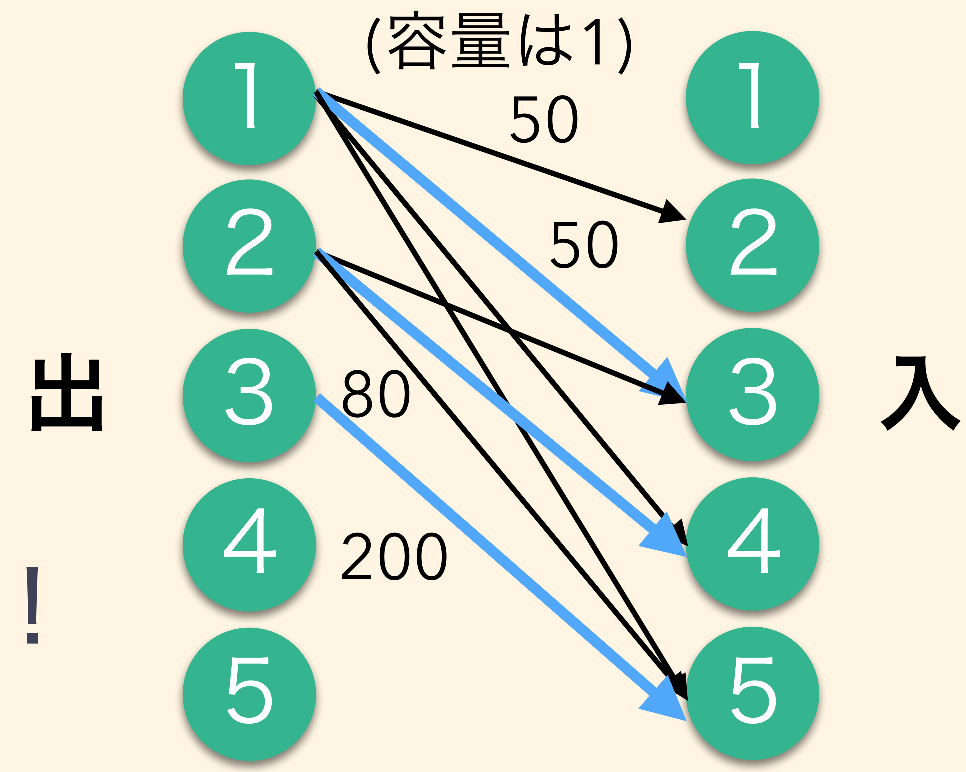
- 最小パス被覆問題は入頂点/出頂点に倍化した二部グラフ上の最大マッチングで解ける (c.f. 蟻本 p.243)





元の問題: 見える体積を最小化

- 最小パス被覆において二部グラフの辺を使う
  ⇔ 出頂点を入頂点のマトリョーシカに格納する
- 二部グラフの辺に (体積) 重みが付いたら…???
  - 重みVの辺を使う ⇔ 体積Vが他のマトリョーシカの中に隠れる
- 隠れる体積を最大化する
  ⇔ 見える体積を最小化する
  → 重み最大マッチングで解ける！

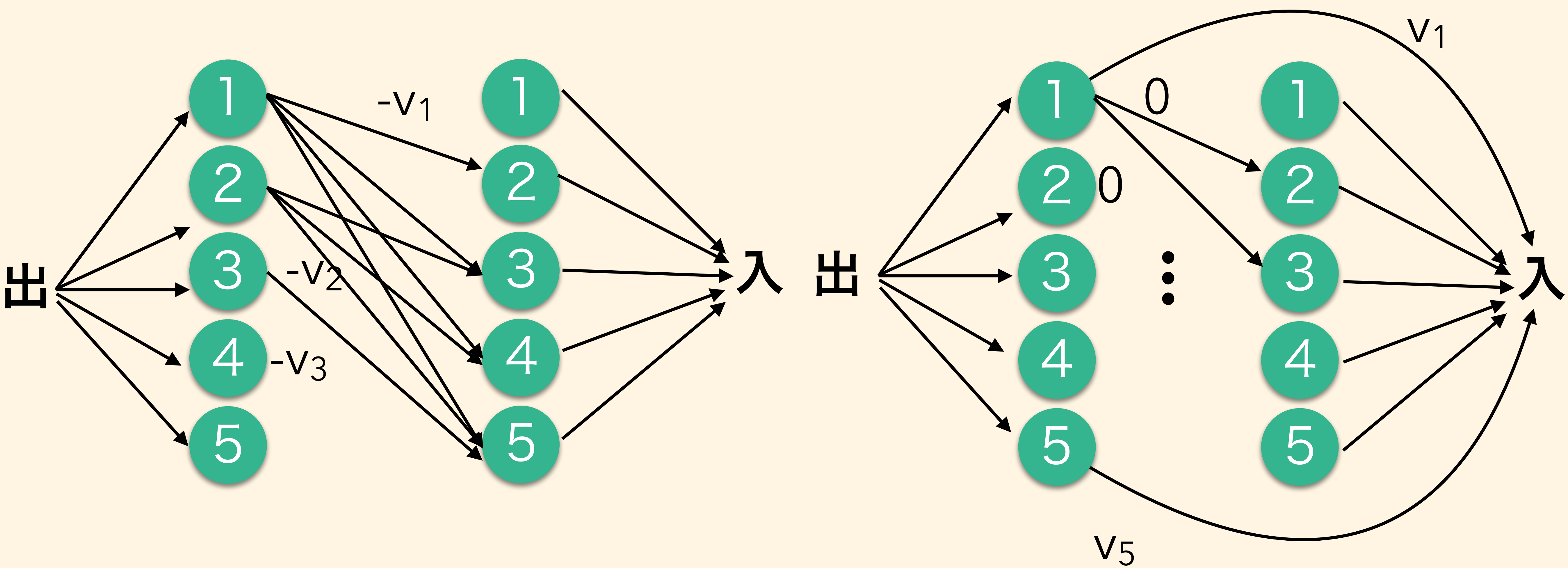


重み最大マッチングの解法

- ハンガリアン法: $O(N^3)$
- 最小費用流: アルゴリズムに依る
  - 頂点数 $V = O(N)$, 辺数 $E = O(N^2)$, 最大流 $F = O(N)$
  - 例えば蟻本 p.203 に載っている $O(F\ E \log V)$ の実装なら、$O(N^3 \log N)$
  - ただし、最大マッチングをしようとすると最初から負の辺があるので注意

重み最大マッチングの解法

- 最小費用流でも以下のようにグラフを変形すると負の辺がなくなる
  - v_i は i 番目のマトリョーシカの体積



Writer 解

- darsein: 134 行, 2844 byte (C++)
- not: ライブラリが分割されておりめんどい (C++)
- tokoharu: 167 行, 3420 bytes (C++)
- torus: 272 行, 6740 bytes (C++)

統計情報

- AC / trying teams
  - 19 / 34 (55.88%)
- First Acceptance
  - 非現役込み: catsatmat (90:50)
  - 現役のみ: catsatmat (90:50)