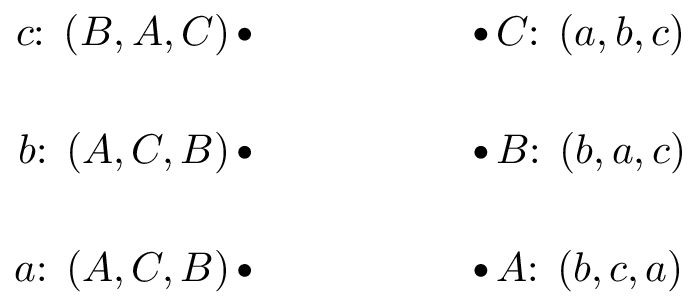
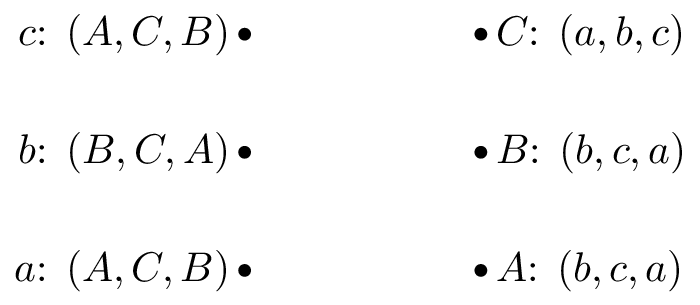
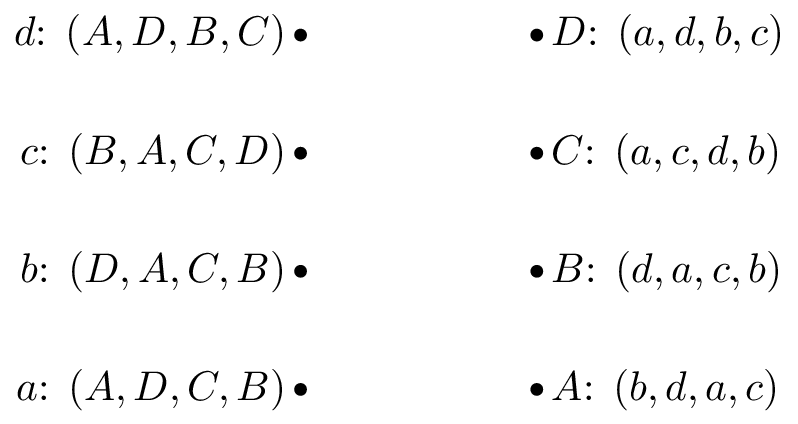
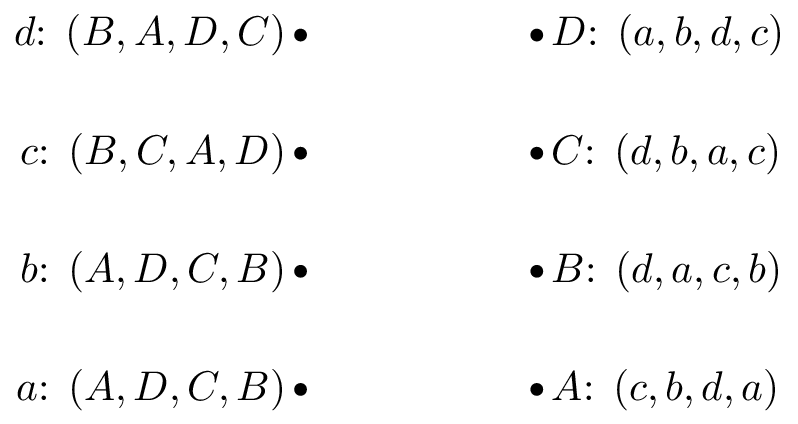
# Homework sheet 5 - Matching games, cooperative games and routing games

1. Obtain stable suitor optimal and reviewer optimal matchings for the matching games shown.
   * Game 1:
   *  **Solution**
   * Following the algorithm:
   * Suitor optimal: Reviewer optimal:
   * Game 2:
   *  **Solution**
   * Following the algorithm:
   * Suitor optimal: Reviewer optimal:
   * Game 3:
   *  **Solution**
   * Following the algorithm:
   * Suitor optimal: Reviewer optimal:
   * Game 4:
   *  **Solution**
   * Following the algorithm:
   * Suitor optimal: Reviewer optimal:
2. Consider a matching game where all reviewers have the same preference list. Prove that there is a single stable matching.

* **Solution**
* Let be the suitor optimal matching (given by the Gale-Shapley algorithm).
* Assume . As is reviewer sub-optimal a subset such that: For all : is worse than . For .
* Consider , as all reviewers have same reference list, let be the reviewer with ''best'' suitor under matching (the matching given by the Gale Shapley algorithm).
* When considering , reviewers outside of have same matching as in . All reviewers in must have a ''better'' matching.
* As all reviewers have the same preference list, cannot be matched thus is not a matching.

1. For the following cooperative games:
   1. Verify if the game is monotonic.
   2. Verify if the game is super additive.
   3. Obtain the Shapley value.

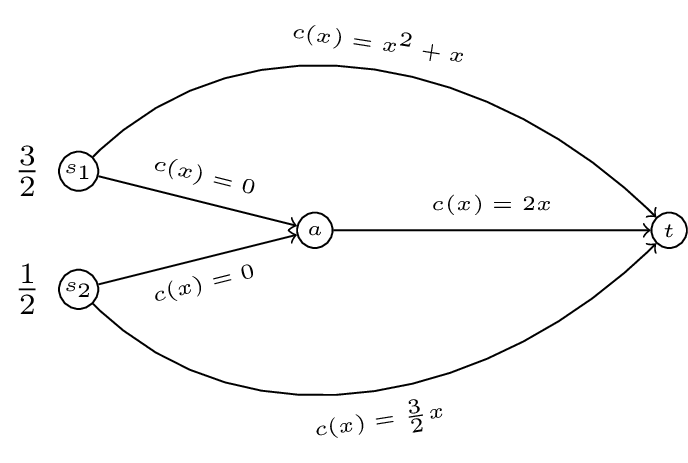
* **Solution**
* Game is monotone but is not super additive: and .
* The Shapley value is .
* **Solution**
* Game is not monotone: . Game is not super additive: .
* The Shapley value is .
* Game is monotone but not super additive:
* The Shapley value is .
* Game is monotone but not super additive:
* The Shapley value is .

1. Prove that the Shapley value has the following properties:
   * Efficiency
   * **Solution**
   * For every permutation we have:
   * taking the mean over all permutations (which is by definition the Shapley value) we have the required result.
   * Null player
   * **Solution**
   * Consider any permutation and a null player . We have . Thus, , as this holds for all the result follows.
   * Symmetry
   * **Solution**
   * Assume that and are symmetric. Given a permutation , let denote the permutation obtained by swapping and .
     + Assume that precedes in , this gives , if we let :
     + and
     + By symmetry .
     + Assume that does not precede in , let . We have:
     + and
     + Since and we have by symmetry and therefore .
   * We have that for all , there is an abvious bijection between all and corresponding thus:
   * as required.
   * Additivity

* **Solution**
* Let be the characteristic function of the game . Following from the definition of additivity it is immediate to note that we have . The result follows.

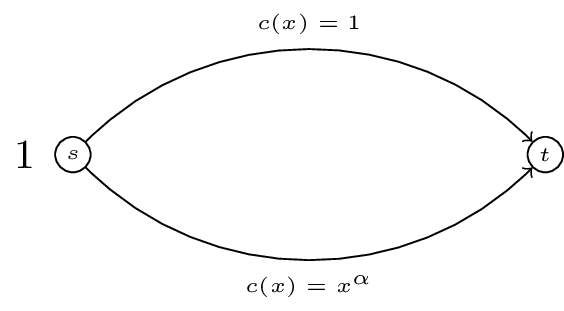
Note that this does not prove that the Shapley value is the only vector that has those properties (it in fact is though).

1. Calculate the Nash flow and the optimal flow for the following routing game:

*  **Solution**
* For the **Nash flow**:
* Let be the traffic along the top arc and the traffic along the bottom arc.
* By definition for commodity 1 we have:
* By definition for commodity 2 we have:
* Solving this later equation gives:
* Substituting this in to the first equation gives:
* which has solution , substituting this in to our expression for gives: .
* For the **Optimal flow** we use the marginal costs:

* We now repeat the above:
* By definition for commodity 1 we have:
* By definition for commodity 2 we have:
* Solving this later equation gives:
* Substituting this in to the first equation gives:
* which has solution , substituting this in to our expression for gives: .

1. For a routing game the 'Price of Anarchy' is defined as:

* For the game shown (a generalisation of "Pigou's example") obtain the PoA as a function of .
* 
* A generalization of Pigou's example
* **Solution**
* Let be the flow along the bottom arc. The Nash flow is immediate:
* giving
* The optimal flow is solves:
* thus
* giving
* Thus:
* It can be shown that the above is a decreasing function in , this implies that as the 'shortcut' gets 'better' (recall that ) the negative effect of selfish behaviour increases.