The algorithm finds the optimal matchings in several steps:

- First it creates an oriented graph - this is a set of vertices and oriented edges (you can imagine this as points and arrows going between them). A vertex can be either a pair (recipient, his related donor) or (altruistic donor) or (bridging donor).
- An edge between vertices means that the donor from the first vertex can give his kidney to the recipient in the second vertex. This is possible only if:
    - The countries are not AUT > IL / IL > AUT
    - The donor's blood group is among recipients *acceptable blood* groups. Here we need to distinguish between:
        - *compatible blood group* for a recipient is a blood group that is compatible with the recipient's blood group according to standard schema where 0 is universal donor and AB is universal recipient
        - *acceptable blood group* for a recipient does not have to be compatible but donor with this blood group can still give his kidney to the recipient (compatible blood groups are always acceptable, but not the other way around)

      For instance in the current matching all patients from Austria had all of 0, A, B, AB as acceptable, on the other hand patients from Israel had only their compatible blood groups as acceptable.

    - Donor and recipient do not have a positive virtual HLA crossmatch - that is recipient does not have antibodies for donor's antigens. It is worth mentioning how we treat broad / split codes of antigens and antibodies in this case. Let's demonstrate it for example. Broad level code A9 has two split levels A23, A24. In this case:

Transplant (antigen code > antibody code)	Virtual Crossmatch
A23 > A23	**positive**
A24 > A24	**positive**
A9 > A9	**positive**
A23 > A24	**negative**
A24 > A23	**negative**
A9 > A23	**positive**
A9 > A24	**positive**
A24 > A9	**positive**
A23 > A9	**positive**

    - It is further possible to set parameters that require
        - compatible blood group for all the exchanges (e.g. A can receive only 0, A)
        - new donor having a better match (either in blood group, *compatibility index* -- see below, or both) than the original donor related to the recipient
        - minimal *compatibility index* for a transplant to be possible
- After creating the edges that determine if a transplant is actually possible, we assign a score to each edge. This is trivially 1 for all edges if the option *use binary scoring* is selected or it is the **compatibility index** of the transplant, which is defined as

compatibility index = 1 x (number of common A alleles) +
3 x (number of common B alleles) +
9 x (number of common DR alleles)

Each number of common alleles can be either 0, 1 or 2. So the compatibility index is a number between 0 - 26. For this we consider only the broad level antigen codes

A	**B**		**DR**
A1	B5	B46	DR1
A2	B7	B47	DR2
A3	B8	B48	DR3
A9	B12	B53	DR4
A10	B13	B59	DR5
A11	B14	B67	DR6
A19	B15	B70	DR7
A28	B16	B73	DR8
A36	B17	B78	DR9
A43	B18	B81	DR10
A80	B21	B82	DR103
A203	B22	B703	DR1403
A210	B27	B2708	DR1404
A2403	B35	B3901	
	B37	B3902	
	B40	B4005	
	B41	B5102	
	B42	B5103	

- After we constructed the graph where vertices are the pairs, altruists or bridging donors and (oriented) edges are the possible transplants, we find all oriented cycles and paths (that can start only with a bridging donor or altruist) in this graph.
- When we have these cycles and paths, we have to combine them in matchings. We can only combine such cycles and sequences together in one matching that do not have common patients in them. We find the set of all *maximal matchings* - a maximal

matching has the property that there is no other cycle/sequence we can add to it that would not use the patients that are already in the matching. Algorithmically this is done in the following way: we construct another graph (totally unrelated to the previous one), it's vertices are all the different paths and cycles and there is an (unoriented) edge between vertices if they do not have common patients. We then look for the maximal connected components in this graph which then correspond to the matchings).

- When we have the set of all possible maximal matchings, we calculate a score for each based on the edge (transplant) scores (= compatibility index). We can then filter or order the matchings based on different criteria like the matching score / number of transplants / number of sequences or cycles in the matching etc.