1. Let $G = K_3$ with vertices x, y, z and edge weights $w(\{x, y\}) = 1$, $w(\{x, z\}) = 2$, and $w(\{y, z\}) = -2$. Starting at x, Dijkstra's algorithm will set $\delta(y) = 1$ and $\delta(z) = 2$ in the first iteration and choose y as the next 'current vertex'. In the next iteration it will mark y as visited and not update $\delta(y)$ after that. Hence at the end of the algorithm $\delta(y)$ will be equal to 1, but the path from x to y via z has length 0.

2. Assume for a contradiciton that there were 2 different minimal spanning trees T_1 and T_2. Let e be an edge contained in T_1 but not in T_2. Let P be the (unique) path in T_2 connecting the two endpoints of e. there is at least one edge f in P with one endpoint in each of the two connected components of $T_1 - e$ (because the first vertex of P lies in one connected component and the last vertex in the other). So $T_1 - e + f$ is a spanning tree, and since T_1 is a minimal spanning tree we conclude that $w(e) \leq w(f)$. Conversely, e connects the two connected components of $T_2 - f$ (because there are paths in $T_2 - f$ connecting the endpoints of e to the enpoints of f). So $T_2 - f + e$ is a spanning tree, and since T_2 is a minimal spanning tree we conclude that $w(f) \leq w(e)$. But this means that $w(e) = w(f)$,so not all edge weights are different.

3. (a) This statement is false. Consider e.g. the graph C_4 with edge weights 1, 1, 1, and 2. Then the path whose only edge is the edge of weight 2 is a shortest path, but after squaring all weights this path will have weight 4, and the path using the other 3 edges will still have weight 3.

(b) This statement is true.

We claim that if T is a MST with respect to some weight function w, and the order in Kruskal's algorithm is chosen such that edges of T come before edges with the same weight that are not contained in T, then the algorithm returns T. Assume that this was not the case, and let e be the first edge outside of T added by the algorithm. Then e connects two different connected components of the previously constructed graph. Let f be an edge of T connecting these two connected components. Since f was not added before e was considered, we know that $w(f) > w(e)$. Note that e connects the two connected components of $T - f$ (because there are paths in $T - f$ connecting the endpoints of e to the enpoints of f). So $T - f + e$ is a spanning tree. But $w(T - f + e) = w(T) - w(f) + w(e) < w(T)$ contradicting the fact that T was a MST.

Since an order for the edges can be used in Kruskal's algorithm for the original weights if and only if it can be used for the squared weights (it only has to be non-decreasing), we conclude that every MST T for the original weights can be the outcome of Kruskal's algorithm for square weights, and therefore is a minimal spanning tree for the squared weights, and conversely, every MST T for the original weights can be the outcome of Kruskal's algorithm for square weights, and therefore is a minimal spanning tree for these weights.

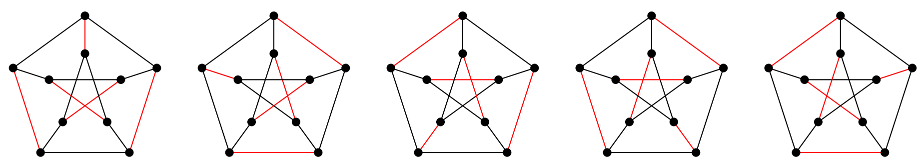
4. For the forward implication, let T be a tree and let M be a perfect matching of T. Let $e = \{u, v\}$ be the edge incident to v contained in M, and note that $M \setminus \{e\}$ is a matching in $T - v$ in which

u is the only unmatched vertex. Hence every connected component of $T - v$ except the one containing u has a perfect matching, and thus has an even number of vertices. The connected component which contains u has a matching in which all vertices except u are matched and therefore has an odd number of vertices. So $o(T - v) = 1$.

For the converse implication, assume that $o(T - v) = 1$ for every vertex v of T. We claim that the set M consisting of all edges such that both connected components of $G - e$ are is a perfect matching. To prove this claim, let v be a vertex of T. If an edge $e = \{u, v\}$ is in M, then u lies in a connected component of $G - e$ (which is also a connected component of $G - v$) with an odd number of vertices. Since $o(T - v) = 1$, at most one edge of M is incident to v. Conversely, if the connected component of $G - v$ containing u has odd size, then the connected component $G - u$ containing v is the union of some even sized connected component of $G - v$ and v, so this has odd size as well. Hence there is an edge of M incident to v. We have thus proved that every vertex v is incident to exactly one edge of M, so M is a perfect matching.

5. (a) Let G be a Δ-regular class 1 graph, and consider a proper edge colouring with Δ edges. every vertex is incident to exactly one edge of each colour, so every colour class forms a perfect matching. Since every edge is contained in some colour class, each edge is contained in some perfect matching, and thus no edge is unmatchable.

(b) By Assignment 1, Question 5a we know that the Petersen graph P is class 2. Below are 5 perfect matchings (all of them are 'rotations' of the first one) of P so that every edge is contained in at least one of them, hence there is no unmatchable edge in P



6. By Kőnig's Theorem (Theorem 2.12 in the lecture notes), the size $m(G)$ of a maximum matching in a bipartite graph is the size $c(G)$ of a minimum vertex cover. By Exercise 1.26 from the lectures, $|V| = c(G) + \alpha(G)$ where $\alpha(G)$ is the size of a maximum independent set.

If M is a matching and I is an independent set then $|M| \leq m(G)$ and $|I| \leq \alpha(G)$, and thus $|M| + |I| \leq m(G) + \alpha(G) = |V|$. Equality holds if and only if $|M| = m(G)$ and $|I| = \alpha(G)$, which (by definition) is the case if and only if M is a maximum matching and I is a maximum independent set.

7. Let P_k denote the path consisting of all edges after Alice's k-th move and let Q_k denote the path consisting of all edges after Bob's k-th move.

Claim: If G has a perfect matching, then Bob has a strategy to win the game.

It is enough to show that Bob is able to choose another edge whenever Alice chooses an edge. Let M be a perfect matching. We will show by induction that Bob can play so that Q_k is an alternating path which starts and ends in a matching edge.

In the first step this can be achieved by choosing a matching edge incident to v.

For the induction step, assume that Q_k is an alternating path which starts and ends in a matching edge. If Alice cannot make another move, then Bob has won and does not need another move. If Alice can make another move, then she must pick an edge $e = \{u, w\}$ where

u is the first or last vertex of Q_k. As Q_k starts and ends in a matching edge we know that $e \notin M$. Moreover, since Q_k is an alternating path, all matching edges incident to vertices in Q_k are contained in Q_k and thus Bob can choose the unique matching edge incident to w to ensure that Q_{k+1} has the desired property.

Claim: If G has no perfect matching, then Alice has a strategy to win the game (and thus Bob cannot have a winning strategy).

It is enough to show that Alice is able to choose another edge whenever Bob chooses an edge. Let M be a maximum matching. We will show by induction that Alice can play so that P_k is the union of two alternating paths (possibly of length 0) both starting in the same unmatched vertex v, and ending in a matching edge unless the length is 0.

In the first step this can be achieved by choosing v to be an unmatched vertex; such a vertex exists because M is not a perfect matching.

For the induction step, assume that P_k is as claimed. If Bob cannot make another move, then Alice has won and does not need another move. If Alice can make another move, then she must pick an edge $e = \{u, w\}$ where u is the last vertex of one of the two alternating paths. As this path ends in a matching edge, or only consists of the unmatched vertex v we know that $e \notin M$. Moreover, v must be matched because otherwise we would have found an augmenting path, contradicting the fact that M is a maximum matching. All matching edges incident to vertices in P_k are contained in P_k and thus Alice can choose the unique matching edge incident to w to ensure that P_{k+1} has the desired property.