CS-E3190 Principles of Algorithmic Techniques

05. Greedy Algorithms – Tutorial Exercise

1. **Maximum weight matching.** Consider a graph $G = (V, E)$, where $|V| = n$, and $|E| = m$. Suppose that the graph is weighted i.e. each edge $e \in E$ is assigned a weight $w(e) \in \mathbb{N}$. Recall that a matching $M \subseteq E$ is a set of edges such that none of them share an endpoint. A matching is called maximal if no edges can be added to it.

   The goal is to design a greedy 2-approximation algorithm for finding a maximum weight matching that runs in $O(m)$.

   (a) Give a greedy algorithm that fits the goal.
   Hint: you can assume that the input graph is given as a list of edges, sorted in a decreasing weights order.

   (b) Prove that this algorithm is correct (returns a matching) and that it gives a 2-approximation of a maximum matching.

   (c) Assuming a sorted array, prove that it runs in time $O(m)$.