1. When we last saw our friends at Algorithmic Pizza, you provided them a greedy algorithm to pick optimal locations to build pizza parlors. One week later, they return to you with a natural follow-up question: in what order should they build the parlors at those locations?

There are $n$ planned pizza parlors to be built. Planned parlor $i$ is described by the number of days $t_i$ it will take to construct it and the expected daily profit $p_i$ it will generate once constructed. Algorithmic Pizza has only one construction crew at their disposal (i.e. only one pizza parlor can be built at a time). Design a greedy algorithm to find a construction order that will maximize the company’s profits. Prove your algorithm’s correctness.
2. Algorithmic Pizza has decided to expand their business by selling and installing premium at-home pizza ovens. Suppose the company has received orders from $n$ customers. Order $i$ is described by an earliest $e_i$ and latest $l_i$ possible date for installation. The installations are a lengthy process, so only one installation can be performed per day.

Having finally learned your methods, the owners of Algorithmic Pizza have been able to come up with an idea for a greedy algorithm on their own for once. They suggest sorting the customers in increasing order of $l_i$, then iterating through them in that order and scheduling each customer on the earliest still-available day in their interval (if there are no days available, then that customer is not served). They believe that this algorithm will maximize the number of customers served, but they aren’t certain. Prove the algorithm’s correctness.

*Hint: Try to prove that there exists an optimal solution that does the same thing as you on the first day. This is an easier approach to this problem than trying to prove that there exists an optimal solution that matches your first decision.*