Editorial - Planet Cruise

There is a constraint that says "You're only allowed to move to planets which has higher x coordinate than the current planet you're in". So the first step is to sort planets by their x coordinates.

Bruteforce

Let's think of the simplest bruteforce solution by simulating the process.

First we're on the 1^{st} planet. Now we have $n-1$ choices. ($n-1$ destination planets). If we pick a planet k to visit, then we can calculate the time using the distance, then we have enough data to find out the cost so far. At the k^{th} planet we will have another $n-k$ choices for the next move. We can write a recursive function to simulate this process. At each step, try out all possible choices keeping track of their costs and find out the minimum.

Time complexity will be $O(n!)$

Dynamic Programming

Note that in the bruteforce solution, the current state of our journy can be explained using two variables. The current position we're in and the current time. There are n possible positions and T possible time units. So there can be only $n * T$ states. All these data strongly suggests that there's a Dynamic Programming solution.

We have already found out about states. so let,

$DP_{i,t}$ = minimum cost to reach the i^{th} planet at t^{th} time unit

Base case: minimum cost to reach the 1^{th} planet at 0^{th} second is 0. (We're already there). Therefore $DP_{0,0} = 0$

At each step, like explained in the bruteforce solution there will be $< n$ choices depending on the position. So thinking backwards, there will be $k-1$ ways to reach the k^{th} planet at a given time t. ($k-1$ planets that could have been the starting point to reach the k^{th} planet)

So the dp equation can be written like this,

$DP_{i,t} = v_{i,t} + \min_{j=0}^{i-1} DP_{j,t-d_{i,j}} + f_i * d_{i,j}$

where,

- f_i is the fuel price at i^{th} planet
- $v_{i,t}$ is the visa cost of i^{th} planet at t^{th} time
- $d_{i,j}$ is the manhattan distance between i^{th} planet and j^{th} planet

Time complexity is num states **x** num transitions. $O(n^2T)$. (This can be reduces to $O(Tn\log n)$ using Convex hull optimization but not required)