A solution is specified by the days on which orders of gas arrive, and the amounts of gas that arrives on those days. In computing the total cost, we will take into account delivery and storage costs, but we can ignore the cost for buying the actual gas, since this is the same in all solutions. (At least all those where all the gas is exactly used up.)

Consider an optimal solution. It must have an order arrive on day 1, and it the next order is due to arrive on day i, then the amount ordered should be $\sum_{j=1}^{i-1} g_j$. Moreover, the capacity requirements on the storage tank say that i must be chosen so that $\sum_{j=1}^{i-1} g_j \leq L$.

What is the cost of storing this first order of gas? We pay g_2 to store the g_2 gallons for one day until day 2, and $2g_3$ to store the g_3 gallons for two days until day 3, and so forth, for a total of $\sum_{j=1}^{i-1}(j-1)g_j$. More generally, the cost to store an order of gas that arrives on day a and lasts through day b is $\sum_{j=a}^{b}(j-a)g_j$. Let us denote this quantity by $S(a, b)$.

Let $OPT(a)$ denote the optimal solution for days a through n, assuming that the tank is empty at the start of day a, and an order is arriving. We choose the next day b on which an order arrives: we pay P for the delivery, $S(a, b-1)$ for the storage, and then we can behave optimally from day b onward. Thus we have the following recurrence.

$$OPT(a) = P + \min_{b>a:\sum_{j=a}^{b-1} g_j \leq L} S(a, b-1).$$

The values of OPT can be built up in order of decreasing a, in time $O(n-a)$ for iteration a, leading to a total running time of $O(n^2)$. The value we want is $OPT(1)$, and the best ordering strategy can be found by tracing back through the array of OPT values.

[1] ex191.358.563