

1. Algorithm Development Task: Inventory Reordering System

Objective : Develop an efficient reordering strategy to ensure no items go out of stock while minimizing total reordering costs.

- Start
- Initialize inputs :- item_id, current_stock, forecasted_demand, reorder_cost_per_unit, batch_size(fixed size in which items can be reordered).
- Calculate shortfall,
$$\text{shortfall} = \text{forecasted_demand} - \text{current_stock}$$
- If shortfall > 0,
Compute minimum reorder quantity,
$$\text{units_order} = \left\lceil \frac{\text{shortfall}}{\text{batch_size}} \right\rceil * \text{batch_size}$$
- Compute total cost of reorder,
$$\text{total_cost} = \text{units_order} * \text{reorder_cost_per_unit}$$
- Add (item_id,units_order) to the reordering list.
- Return the final reordering plan.
- Stop
- For Example,

item_id	current_stock	forecasted_demand	reorder_cost_per_unit	batch_size
1	50	80	5	10
2	200	150	3	20
3	30	60	4	15

=>item_id=1,

shortfall = 80-50=30

units_order= $\left\lceil \frac{30}{10} \right\rceil * 10 = 30$

=>item_id=2,

shortfall = 150-200=-50

units_order=0

=>item_id=3,

shortfall = 60-30=30

units_order= $\left\lceil \frac{30}{10} \right\rceil * 10 = 30$

- Output Rerordering plan,
[(1,30) , (3,30)]