LEAST COST METHOD CODE:-

function [cost, allocation] = least\_cost\_method(cost\_matrix, supply, demand)

[m, n] = size(cost\_matrix);

allocation = zeros(m, n);

while any(supply > 0) & any(demand > 0)

min\_cost = min(cost\_matrix(:));

[row, col] = find(cost\_matrix == min\_cost, 1);

if supply(row) < demand(col)

allocation(row, col) = supply(row);

demand(col) = demand(col) - supply(row);

supply(row) = 0;

else

allocation(row, col) = demand(col);

supply(row) = supply(row) - demand(col);

demand(col) = 0;

end

cost = sum(sum(allocation .\* cost\_matrix));

end

end

// Example usage:

cost\_matrix = [3 6 4; 7 2 8; 5 9 1];

supply = [20; 30; 40];

demand = [25 35 30];

[cost, allocation] = least\_cost\_method(cost\_matrix, supply, demand);

disp(cost);

disp(allocation);