4. local search

a) 1. we need an initail state, so assign the tiles in order to a coefficient
2. we need to define our neighborhood
3. compare and prioritize solutions / Heuristic

Initial state

Assign randomly each tile to a position
while total != y:
if the total is > y:
swap the highest products tile with the tile-1 value

how to expand

if total < y:
swap the lowest Products tile with n+1 value

prioritize values that are closer to y

Heuristic

b.) no, a local search can only provide a local optimum solution, which is not the same as a global optimal solution. It might be optimal for its given neighborhoods, but not necessarily for the whole solution. if every way it tries next is worse it will claim to have the optimal, which isn't always the case.