🔥 Classic "Sequence with Transition Constraint" DP Problems

🏠 Paint / Coloring Problems

1. **Paint House I (LeetCode 256)**
   - Each house can be painted with 3 colors.
   - Adjacent houses can't have the same color.
   - `dp[i][color] = min cost up to house i with color .`
2. **Paint House II (LeetCode 265)**
   - Generalization with `k` colors.
   - Need optimization for O(n·k) → O(n·k) with min-tracking.
3. **Paint Fence (LeetCode 276)**
   - No more than 2 adjacent fences can have the same color.

🥷 Training / Task Scheduling

4. **Ninja's Training / Geek's Training (GFG)**
   - Pick 1 of 3 activities per day, no two consecutive days same activity.
5. **Activity Selection with Constraints**
   - Each day choose one of k activities, can't repeat yesterday's.
   - Sometimes adds cooldown (like "can't pick task again for 2 days").
6. **House Robber I (LeetCode 198)**
   - Linear houses, can't rob adjacent.
   - `dp[i] = max(dp[i-1], dp[i-2] + val[i]) .`
7. **House Robber II (LeetCode 213)**
   - Houses in a circle (first and last also adjacent).
8. **House Robber III (LeetCode 337)**
   - Houses in a tree, can't rob parent & child simultaneously.
   - Tree DP version.

⏰ Scheduling / Job Problems

9. **Maximum Alternating Subsequence Sum (LeetCode 1911)**
    - Pick elements with alternating +/− contribution.
    - Similar to "choose, but not same mode as before".
10. **Stock Buy and Sell with Cooldown (LeetCode 309)**
    - State machine DP (buy, sell, cooldown).
    - Transition restricted by last action.
11. **Stock Buy and Sell with Fee (LeetCode 714)**
    - Variation with fee deduction on sell.
12. **Weighted Job Scheduling (LeetCode 1235)**
    - Jobs with start & end times, can't overlap.
    - Transition depends on compatibility with previous job.

📈 Subsequence/Sequence Constraints

13. **Longest Alternating Subsequence (GFG / LeetCode 376)**
    - Difference between consecutive elements must alternate sign.
14. **Delete and Earn (LeetCode 740)**
    - Pick a number, can't pick adjacent numbers (similar to House Robber).
15. **Frog Jump (GFG / Codeforces style)**
    - Frog can jump 1 or 2 steps, minimize cost.
    - Transition depends on previous step.
16. **Frog Jump with K Distance**
    - Generalization: can jump up to K steps.

🎭 Expression / Partition Problems

17. **Palindrome Partitioning II (LeetCode 132)**
    - Cut string into palindromes, minimize partitions.
    - Transition depends on previous partition.
18. **Evaluate Boolean Expression to True (GFG)**
    - Partition expression at operators, combine results.
    - State: `(i, j, isTrue)` .
19. **Burst Balloons (LeetCode 312)**

- Interval DP, last balloon to burst in range matters.

20. **Matrix Chain Multiplication**
    - Order of multiplication restricted by partitions.

⚡ Key Patterns

- **Paint House / Ninja Training** → *Pick one of k options, can't repeat last.*
- **House Robber / Delete and Earn** → *Can't take adjacent items.*
- **Stock DP** → *Action restricted by last action (buy/sell/cooldown).*
- **Partition DP** → *Split problem at some k, transition depends on split.*