# Binary Search – Core Patterns Summary

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Pattern | Core Idea | Mental Idea | Typical Scenarios |
| 1 | Classic Binary Search (Value Search) | Divide a sorted range in half each iteration to locate a target element. | I can repeatedly eliminate half of the space. | Search in Sorted Array, Find Element in Rotated Array |
| 2 | Lower Bound (First Occurrence) | Find the first index where element ≥ target. | I want the leftmost position where this condition becomes true. | Binary Search Variant, Search Insert Position |
| 3 | Upper Bound (Last Occurrence) | Find the last index where element ≤ target. | I want the rightmost position where this condition remains true. | Count of Element, Last Occurrence in Sorted Array |
| 4 | Binary Search on Answer (Search Space Reduction) | Search not in an array, but on the range of possible answers (minimize/maximize something). | I can’t directly find the answer, but I can check if a candidate works and binary search over all possibilities. | Aggressive Cows, Capacity to Ship Packages, Minimum Days to Make M Bouquets |
| 5 | Binary Search on Real Numbers / Precision Search | Apply binary search on continuous values until desired precision is reached. | The function is monotonic, so I can narrow down until I reach a small enough error margin. | Square Root Calculation, Koko Eating Bananas (precision version) |
| 6 | Rotated / Shifted Array Search | Handle partially sorted arrays (rotation point unknown). | One side of the array is sorted — decide which half to search. | Search in Rotated Sorted Array, Find Minimum in Rotated Sorted Array |
| 7 | Binary Search in 2D Matrix | Treat 2D matrix as a linear sorted space. | Flatten the matrix or binary search row + column logically. | Search a 2D Matrix I/II |
| 8 | Binary Search on Functions (Monotonic Predicate) | Apply binary search where the condition flips from false → true in sorted order. | Find the smallest/largest value where a boolean condition switches. | First Bad Version, Min Speed to Arrive on Time |
| 9 | Peak / Local Optimum Search (Binary Search on Shape) | Search for a local maximum/minimum in unimodal or bitonic arrays. | Compare neighbors to decide which half contains the peak. | Find Peak Element, Bitonic Array Peak |
| 10 | Binary Search with Answer Validation (Decision Problems) | After finding candidate, verify or adjust boundaries by conditions. | Once I have a candidate, I still validate if it truly satisfies the problem constraints. | Allocate Books, Split Array Largest Sum |