**Big O Notation:**  
Describes how time grows with input size.

* Best/Average/Worst Case:
  + **Linear Search**: O(1), O(n), O(n)
  + **Binary Search**: O(1), O(log n), O(log n)

**Implementation:**

java

// Linear Search

for (Product p : productList) {

if (p.productName.equals("Shoes")) {

return p;

}

}

java

// Binary Search

int binarySearch(Product[] arr, String key) {

int low = 0, high = arr.length - 1;

while (low <= high) {

int mid = (low + high) / 2;

int cmp = arr[mid].productName.compareTo(key);

if (cmp == 0) return mid;

else if (cmp < 0) low = mid + 1;

else high = mid - 1;

}

return -1;

}

**Comparison:** Binary search is much faster but needs sorted input.