**Week-1: (Module 2)**

**Date Structures and Algorithms**

**Exercise 2: E-commerce Platform Search Function**

**Product.java**

public class Product {

    int productId;

    String productName;

    String category;

    public Product(int productId, String productName, String category) {

        this.productId = productId;

        this.productName = productName;

        this.category = category;

    }

    @Override

    public String toString() {

        return "Product{" + productId + ", '" + productName + "', '" + category + "'}";

    }

}

**LinearSearch.java**

public class LinearSearch {

    public static Product search(Product[] products, String name) {

        for (Product p : products) {

            if (p.productName.equalsIgnoreCase(name)) {

                return p;

            }

        }

        return null;

    }

}

**BinarySearch.java**

public class BinarySearch {

    public static Product search(Product[] products, String name) {

        int low = 0;

        int high = products.length - 1;

        while (low <= high) {

            int mid = (low + high) / 2;

            int cmp = products[mid].productName.compareToIgnoreCase(name);

            if (cmp == 0) return products[mid];

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

            else high = mid - 1;

        }

        return null;

    }

}

**Main.java**

import java.util.Arrays;

import java.util.Comparator;

public class Main {

    public static void main(String[] args) {

        int n = 100000;

        Product[] products = new Product[n];

        for (int i = 0; i < n; i++) {

            products[i] = new Product(i, "Product" + i, "Category" + (i % 10));

        }

        String target = "Product99999";

        long startTime = System.nanoTime();

        Product result1 = LinearSearch.search(products, target);

        long endTime = System.nanoTime();

        System.out.println("Linear Search Result: " + (result1 != null ? result1 : "Not Found"));

        System.out.println("Time taken (Linear): " + (endTime - startTime) + " ns");

        Arrays.sort(products, Comparator.comparing(p -> p.productName.toLowerCase()));

        startTime = System.nanoTime();

        Product result2 = BinarySearch.search(products, target);

        endTime = System.nanoTime();

        System.out.println("\nBinary Search Result: " + (result2 != null ? result2 : "Not Found"));

        System.out.println("Time taken (Binary): " + (endTime - startTime) + " ns");

    }

}

Output:



Conclusion:

The results shows that:

* **Linear Search**, while straightforward and not requiring sorted data, was significantly slower for large datasets.
* **Binary Search**, after sorting the array once, provided much faster search times due to its logarithmic time complexity.