**Algorithms and Data Structures**

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

**Product.java:**

public class Product {

int productId;

String productName;

String category;

public Product(int id, String name, String category) {

this.productId = id;

this.productName = name;

this.category = category;

}

public String toString() {

return "Product ID: " + productId + ", Name: " + productName + ", Category: " + 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.\*;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter number of products: ");

int n = sc.nextInt();

sc.nextLine();

Product[] products = new Product[n];

for (int i = 0; i < n; i++) {

System.out.println("\nEnter details for Product " + (i + 1));

System.out.print("Product ID: ");

int id = sc.nextInt();

sc.nextLine();

System.out.print("Product Name: ");

String name = sc.nextLine();

System.out.print("Category: ");

String category = sc.nextLine();

products[i] = new Product(id, name, category);

}

System.out.print("\nEnter product name to search: ");

String searchName = sc.nextLine();

System.out.println("\n--- Linear Search ---");

Product foundLinear = LinearSearch.search(products, searchName);

System.out.println(foundLinear != null ? "Product Found: " + foundLinear : "Product Not Found.");

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

System.out.println("\n--- Binary Search ---");

Product foundBinary = BinarySearch.search(products, searchName);

System.out.println(foundBinary != null ? "Product Found: " + foundBinary : "Product Not Found.");

System.out.println("\n--- Time Complexity Analysis ---");

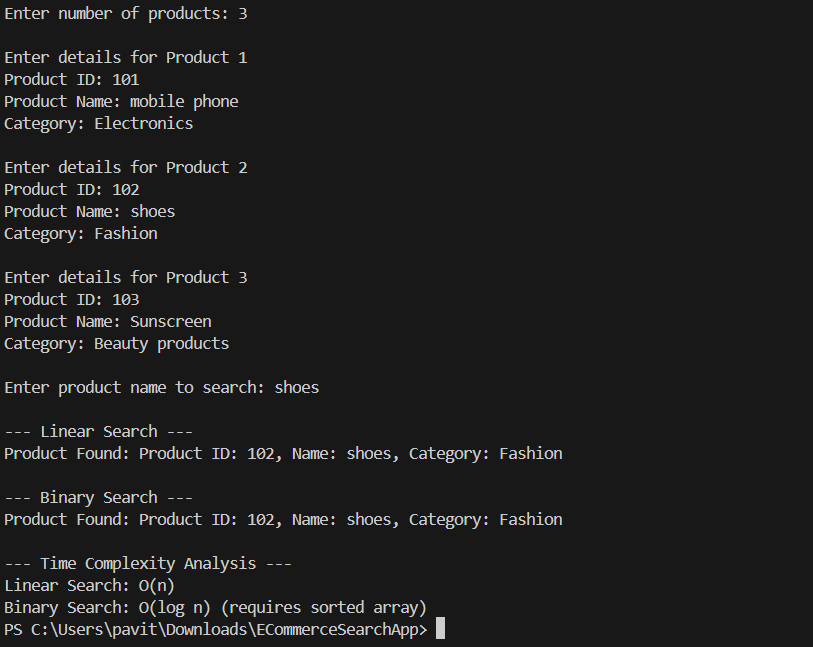
System.out.println("Linear Search: O(n)");

System.out.println("Binary Search: O(log n) (requires sorted array)");

}

}

**Output:**

****