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

**Scenario:**

You are working on the search functionality of an e-commerce platform. The search needs to be optimized for fast performance.

1. **Understand Asymptotic Notation:**

**Big O notation**

**Big O notation** is a mathematical way to describe the **efficiency of an algorithm** as the input size increases. It focuses on the **growth rate** of an algorithm rather than exact execution time, helping developers understand how scalable or efficient their code is.

* It **abstracts away constants and lower-order terms**, giving a high-level view of performance.
* Helps in **comparing algorithms**.
* Useful in predicting **how the program behaves with large inputs**.

**Linear Search**

* **Description**: Go through each element one by one until the target is found.

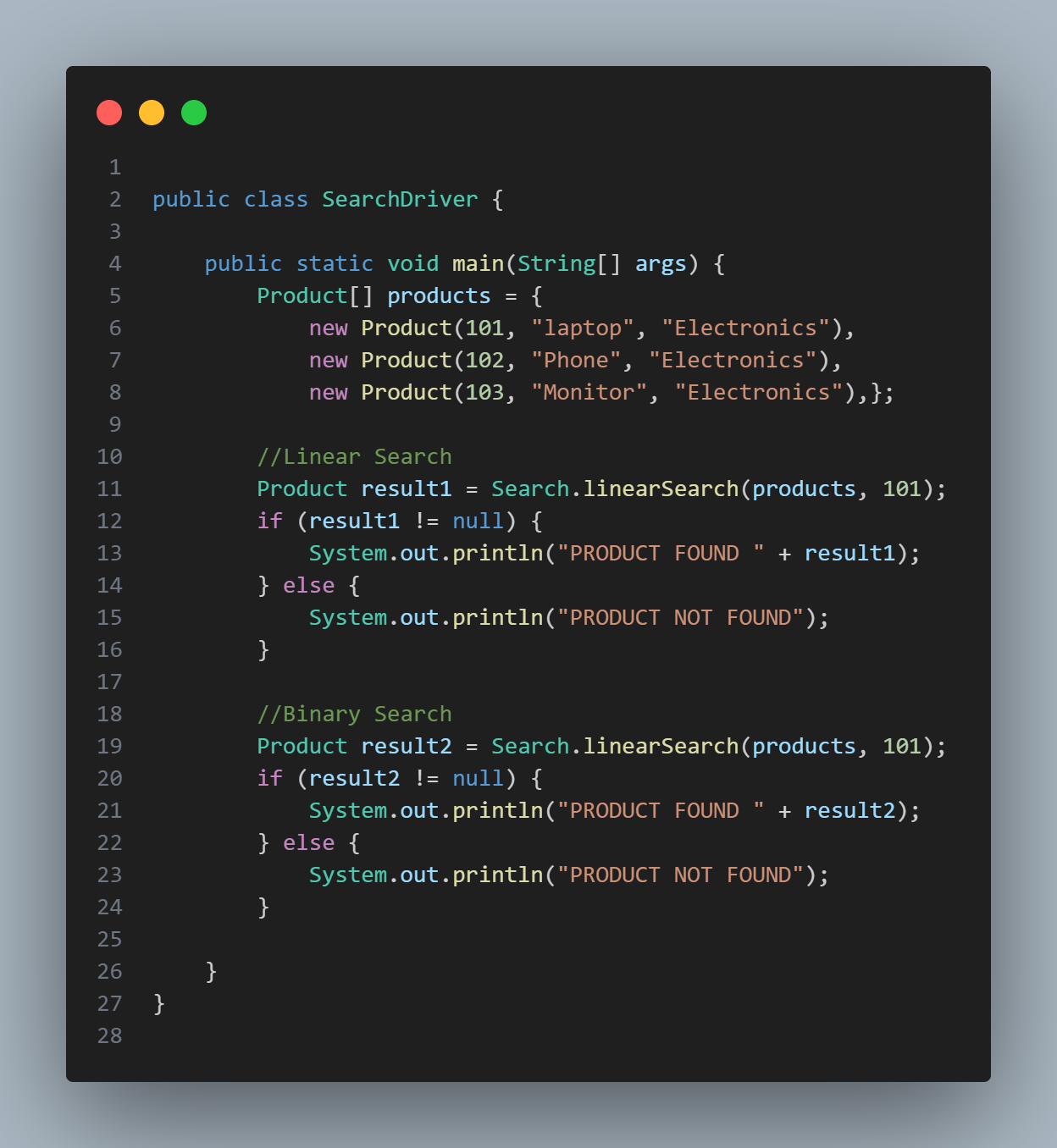
| **Case** | **Time Complexity** | **When It Happens** |
| --- | --- | --- |
| **Best** | O(1) | Target is the **first** element. |
| **Average** | O(n) | Target is somewhere in the **middle**. |
| **Worst** | O(n) | Target is at the **end** or **not present**. |

**Binary Search (works only on sorted data)**

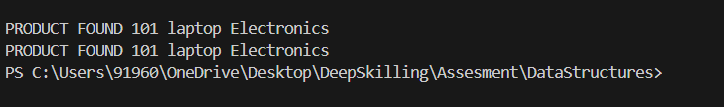
* **Description**: Repeatedly divide the search space in half.

| **Case** | **Time Complexity** | **When It Happens** |
| --- | --- | --- |
| **Best** | O(1) | Target is at the **middle** on the first try. |
| **Average** | O(log n) | Target is found after several divisions. |
| **Worst** | O(log n) | Target is **not present** or found last. |





**OUTPUT**

****