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

**Step-1:**

**Understand Asymptotic Notation:**

**Big O notation**

* Big O notation is a mathematical notation used to describe the performance or complexity of an algorithm, specifically the upper bound of the growth rate of the algorithm's time or space requirements as the input size increases.
* Big O notation provides an upper limit on the time taken by an algorithm in terms of the size of the input, focusing on the worst-case scenario.
* It describes the asymptotic behavior (order of growth of time or space in terms of input size) of a function, not its exact value.
* We can find complexity of codes by mathematical functions using Big O.

Also, It helps in Analyzing Algorithms by

* Showing how algorithms performance changes with input size.Ex-

O(n2) will be slower than O(n) as n increases.

* It helps us to decide which algorithm to follow before even executing or implementing the codes.
* Helps lot in decision making

For Search Operations i.e., linear search and binary search we will discuss Best, Average and Worst Cases

**Linear Search**:

Best Case: O(1)-if it is first element

Average Case: O(n/2) 🡪 O(n)  
 Worst Case: O(n)-last element/not there

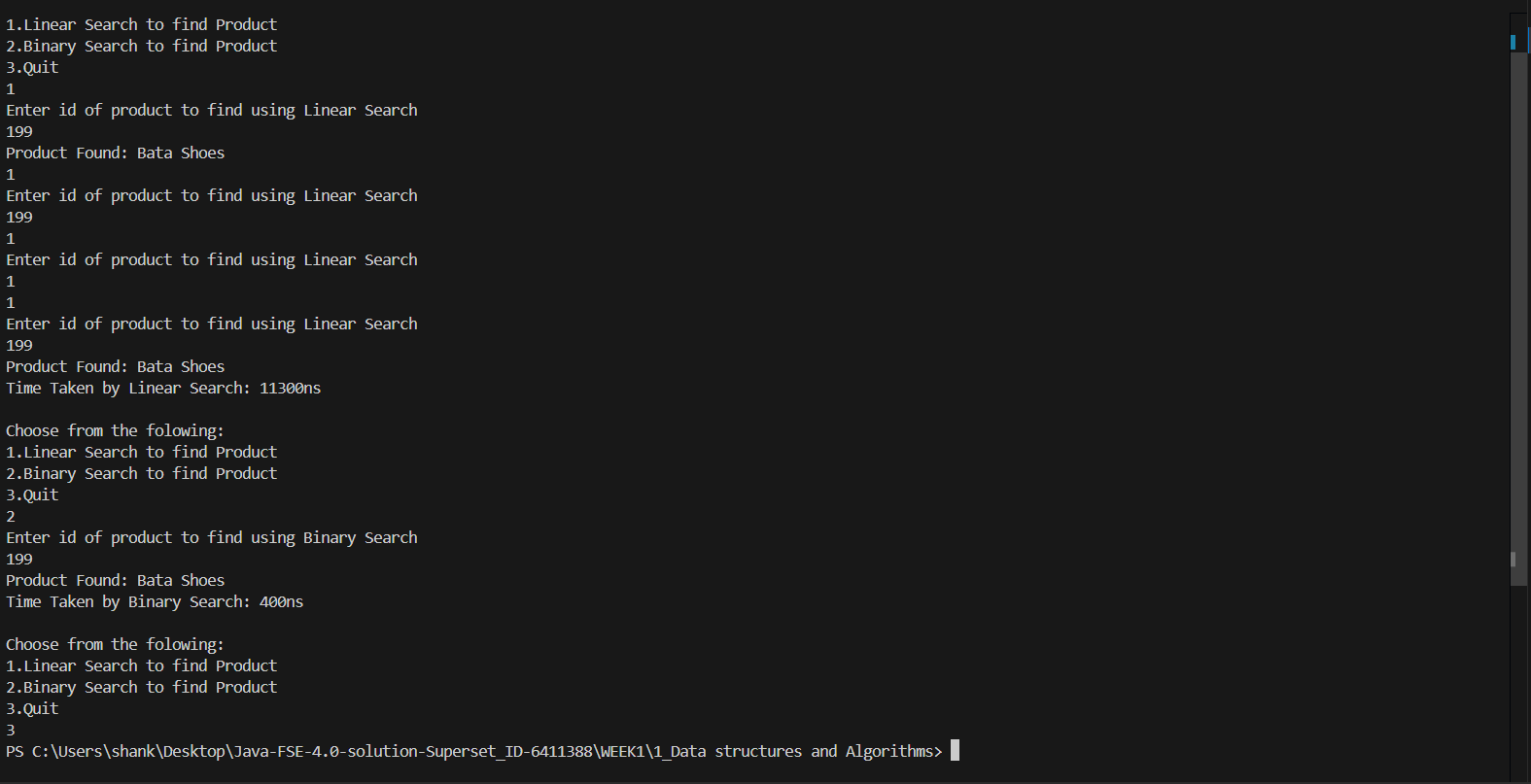
**Binary Search**:

Best Case: O(1)-if it is middle element

Average Case: O(log n)  
 Worst Case: O(log n)

**Refer repository for Implementation(code) & output**

**Analysis:**

****

In above image, we can conclude that Linear Search takes much time than Binary Search

i.e., linear (11300 ns) > binary(400 ns)

**Binary Search** is more suitable **when**:

* The product catalog is **large**
* The products are **already sorted** or can be sorted once and reused.

**Linear Search** is more suitable:

* For **small datasets**
* When the data is **unsorted** and you need a quick one-time lookup.

But in real-world, We need to **use Binary Search** which is more efficient compared to linear why because current systems are critical and large.