**Introduction to Data Structures and Algorithms (DSA)**

**What Are Data Structures?**

Data structures are ways of organizing and storing data so they can be accessed and worked with efficiently.

**Daily Life Analogy:**  
Imagine a cupboard. You can arrange clothes in drawers, hangers, or shelves depending on what’s most efficient for finding and using them.

**What Are Algorithms?**

Algorithms are step-by-step instructions to solve a problem or perform a task.

**Daily Life Analogy:**  
A recipe for making tea is an algorithm:

Boil water,

Add tea leaves,

Add sugar and milk,

Stir and serve

**Why Learn DSA?**

Helps you solve problems efficiently

Forms the backbone of programming logic

Essential for technical interviews and app development

**Core Concepts with Real-Life Analogies and Code**

**1. Stack = Stack of Plates (LIFO)**

**Analogy:** The last plate placed on top of the stack is the first to be removed.

**Real-Life Example:** Undo action in a text editor.

**JavaScript Example:**

*let stack = [];*

*stack.push("Undo Step 1");*

*stack.push("Undo Step 2");*

*stack.push("Undo Step 3");*

*console.log(stack.pop()); // Output: Undo Step 3*

**2. Queue = Line at a Ticket Counter (FIFO)**

**Analogy:** The first person in line is the first to get a ticket.

**Real-Life Example:** Print queue at a cyber cafe.

**JavaScript Example:**

*let queue = [];*

*queue.push("Person A");*

*queue.push("Person B");*

*queue.push("Person C");*

*console.log(queue.shift()); // Output: Person A*

**3. Pseudocode = Cooking Recipe**

**Analogy:** Like a recipe, pseudocode gives clear instructions without actual coding syntax.

**Real-Life Example:** Steps to prepare a sandwich.

**Pseudocode:**

SET bread = 2 slices

ADD butter

ADD cheese

PUT slices together

SERVE sandwich

**JavaScript Equivalent:**

*let bread = 2;*

*console.log("Add butter");*

*console.log("Add cheese");*

*console.log("Put slices together");*

*console.log("Serve sandwich");*

**4. Flowchart = Road Map with Signs**

**Analogy:** A flowchart shows different paths depending on conditions, like a GPS guiding you.

**Real-Life Example:** Choosing an outfit based on the weather.

**Flowchart Decision:**

*IF it is raining THEN*

*Wear raincoat*

*ELSE*

*Wear T-shirt*

**JavaScript Example:**

*let weather = "rainy";*

*if (weather === "rainy") {*

*console.log("Wear a raincoat");*

*} else {*

*console.log("Wear a T-shirt");*

*}*

**5.Linear Search?**

**Linear search** is a simple method to find an item in a list. You start from the first item and check each one until you find what you're looking for (or reach the end of the list).

**Real-Life Analogy (Daily Activity):**

Activity: Imagine you're looking for your friend's name in a handwritten guest list.

You start from the top and check each name one by one until you find your friend.

**JavaScript Code Example:** Linear Search

Let’s write a function that searches for a fruit in a list:

*function linearSearch(array, target) {*

*for (let i = 0; i < array.length; i++) {*

*if (array[i] === target) {*

*return `Found '${target}' at position ${i}`;*

*}*

*}*

*return `'${target}' not found in the list.`;*

*}*

*// Example usage*

*let fruits = ["Apple", "Mango", "Banana", "Orange"];*

*console.log(linearSearch(fruits, "Banana")); // Found 'Banana' at position 2*

*console.log(linearSearch(fruits, "Grapes")); // 'Grapes' not found in the list.*

**Exercise doing in class**

1. **Create a stack and add the names of your 3 favorite fruits. Remove the last fruit and display the remaining ones.**

**2.Simulate a queue for students entering the classroom. Add 4 names, then remove the first 2. Print the final queue.**

**3.Write your own pseudocode for preparing a cup of coffee. Then convert it into simple JavaScript console log statements.**

1. **Create a decision-based flow using if-else to determine what meal to eat based on the time of day (morning, afternoon, evening).**

**5.You have a list of books on a table:  
["Math", "Science", "English", "History", "Art"]**

**You want to search for the book titled "History" using linear search.  
Write a program to find the book and print its position (index).**

**Answers**

1. **Stack – Favorite Fruits**

Solution:

*let fruitStack = [];*

*fruitStack.push("Mango");*

*fruitStack.push("Banana");*

*fruitStack.push("Apple");*

*fruitStack.pop(); // Removes "Apple"*

*console.log("Remaining fruits:", fruitStack);*

*// Output: Remaining fruits: [ 'Mango', 'Banana' ]*

1. **Queue – Students Entering the Classroom**

Solution:

*let studentQueue = [];*

*studentQueue.push("Alice");*

*studentQueue.push("Bob");*

*studentQueue.push("Charlie");*

*studentQueue.push("Diana");*

*studentQueue.shift(); // Removes "Alice"*

*studentQueue.shift(); // Removes "Bob"*

*console.log("Final queue:", studentQueue);*

*// Output: Final queue: [ 'Charlie', 'Diana' ]*

1. **Pseudocode & JavaScript – Making a Cup of Coffee**

Solution:

Psql:

*BOIL water*

*ADD coffee powder to cup*

*POUR boiled water into cup*

*ADD sugar and milk*

*STIR well*

Javascript:

*console.log("Boil water");*

*console.log("Add coffee powder to cup");*

*console.log("Pour boiled water into cup");*

*console.log("Add sugar and milk");*

*console.log("Stir well");*

1. **Flowchart Logic – Meal Based on Time of Day**

Solution:

*let timeOfDay = "afternoon"; // Try "morning", "afternoon", or "evening"*

*if (timeOfDay === "morning") {*

*console.log("Eat breakfast");*

*} else if (timeOfDay === "afternoon") {*

*console.log("Eat lunch");*

*} else if (timeOfDay === "evening") {*

*console.log("Eat dinner");*

*} else {*

*console.log("Invalid time of day");*

*}*

1. **JavaScript Code:**

**Solution:**

*function linearSearch(books, targetBook) {*

*for (let i = 0; i < books.length; i++) {*

*if (books[i] === targetBook) {*

*return ` ${targetBook}' found at position ${i}`;*

*}*

*}*

*return `❌ ${targetBook}' not found in the list.`;*

*}*

*let bookList = ["Math", "Science", "English", "History", "Art"];*

*console.log(linearSearch(bookList, "History"));*

*console.log(linearSearch(bookList, "Biology"));*