**Practical 1: Linear Search (C Language)**

**Theory Section**

Linear Search is the most basic searching algorithm. It works by checking each element of the array one by one until the key is found or the end of the array is reached. If the key is found, the index is returned; otherwise, -1 is returned. This method does not require the array to be sorted.  
  
Steps:  
1. Start from index 0.  
2. Compare each element with the key.  
3. If found, return index.  
4. If not found till end, return -1.  
  
Best Case: O(1) — key is at the beginning.  
Worst Case: O(n) — key is at the end or not present.

**Code Section (C Implementation with Full Comments)**

#include <stdio.h>  
  
// Function to perform linear search  
int linearSearch(int arr[], int n, int key) {  
 // Loop through each element of the array  
 for (int i = 0; i < n; i++) {  
 // If current element matches the key, return index  
 if (arr[i] == key)  
 return i;  
 }  
 // If key not found, return -1  
 return -1;  
}  
  
int main() {  
 int n, key;  
  
 // Ask user for the number of elements  
 printf("Enter the number of elements: ");  
 scanf("%d", &n);  
  
 int arr[n];  
  
 // Input elements into the array  
 printf("Enter %d elements:\n", n);  
 for (int i = 0; i < n; i++)  
 scanf("%d", &arr[i]);  
  
 // Ask user for the element to search  
 printf("Enter the element to search: ");  
 scanf("%d", &key);  
  
 // Call the linearSearch function  
 int result = linearSearch(arr, n, key);  
  
 // Print result based on return value  
 if (result != -1)  
 printf("Element found at index %d\n", result);  
 else  
 printf("Element not found in the array.\n");  
  
 return 0;  
}