Assume that an array A with n elements was sorted in an ascending order, but two of its elements swapped their positions by a mistake while maintaining the array. Write a code to identify the swapped pair of elements and their positions in the asymptotically best possible time. [Assume that all given elements are distinct integers.

CODE:

#include <stdio.h>

void findSwappedPair(int arr[], int n) {

int firstIndex = -1, secondIndex = -1;

for (int i = 0; i < n - 1; i++) {

if (arr[i] > arr[i + 1]) {

if (firstIndex == -1) {

firstIndex = i;

} else {

secondIndex = i + 1;

break;

}

}

}

if (secondIndex == -1) {

secondIndex = firstIndex + 1;

}

printf("Swapped elements: %d and %d\n", arr[firstIndex], arr[secondIndex]);

printf("Positions: %d and %d\n", firstIndex, secondIndex);

}

int main() {

int n;

printf("Enter the number of elements in the array: ");

scanf("%d", &n);

int arr[n];

printf("Enter the elements of the array in ascending order with two elements swapped:\n");

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

scanf("%d", &arr[i]);

}

findSwappedPair(arr, n);

return 0;

}