pseudo-code of the algorithm for the `IsSumFound` function:######

```plaintext
Function IsSumFound(arr: Array of int, n: int, sum: int) -> (int, int):
    Initialize two pointers, left = 0 and right = n - 1

    While left < right:
        currentSum = arr[left] + arr[right]

        If currentSum == sum:
            Return 1, left, right  // Sum found, return 1 and the indexes

        If currentSum < sum:
            Increment left  // Move towards larger elements
        Else:
            Decrement right  // Move towards smaller elements

    Return 0, -1, -1  // Sum not found, return 0 and -1 for indexes
```

############ explanation of the algorithm: ############

1. Initialize two pointers, `left` and `right`, at the beginning and end of the sorted array, respectively.

2. Enter a loop that continues as long as `left` is less than `right`.

3. Calculate the current sum of the elements at positions `left` and `right`.

4. If the current sum is equal to the target sum, return `1` along with the indexes `left` and `right`.

5. If the current sum is less than the target sum, move the `left` pointer to the right, increasing the sum.

6. If the current sum is greater than the target sum, move the `right` pointer to the left, decreasing the sum.

7. If the loop finishes without finding a valid sum, return `0` along with `-1` for the indexes to indicate that the sum was not found.

############ implementation ############

```
#include <stdio.h>

int IsSumFound(int arr[], int n, int sum, int *index1, int *index2)
{
    int left = 0;
    int right = n - 1;

    while (left < right)
    {
        int currentSum = arr[left] + arr[right];

        if (currentSum == sum)
        {
            *index1 = left;
            *index2 = right;
            return 1; // Sum found
```

```
        }

        if (currentSum < sum)
        {
            left++;
        }
        else
        {
            right--;
        }
    }

    return 0; // Sum not found
}

int main()
{
    int arr[] = {2, 4, 7, 12, 14};
    int n = sizeof(arr) / sizeof(arr[0]);
    int sum = 21;
    int index1, index2;

    int result = IsSumFound(arr, n, sum, &index1, &index2);

    if (result == 1)
    {
        printf("Sum found: %d + %d = %d\n", arr[index1], arr[index2], sum);
    }
    else
    {
        printf("Sum not found.\n");
    }

    return 0;
}
```