```c
#include <stdio.h>

#define N       5
#define S       5
#define BOOLEAN unsigned short
#define TRUE    1
#define FALSE   0
#define ZERO    0
#define ONE     1

//-----------------------------------------------------------------------------------
//                                              SumInArray
//                                              ----------
//
// General      : Checks if there are two numbers whose sum is equal to
//                the number entered as "sum".
//
// Parameters   :
//          array[] - array (int)
//          size    - array size (unsigned int)
//          sum     - The amount required (int)
//
// Return value : Counter (int).
//
//-----------------------------------------------------------------------------------
// Programmer : Cohen Idan
// Student No : 211675038
// Date       : 21.10.2019
//-----------------------------------------------------------------------------------
BOOLEAN SumInArray(int array[], unsigned int size, double sum)
{
    BOOLEAN answer = FALSE;
    unsigned int temp_size = size,
                 count;
    double temp_sum;

        (count = ZERO; count < temp_size && !answer;)
    {
        temp_sum = array[count] + array[temp_size];
        answer = (temp_sum == sum) ? TRUE : FALSE;
        count += (temp_sum < sum) ? ONE : ZERO;
        temp_size -= (temp_sum > sum) ? ONE : ZERO;
    }

           (answer);
}


//-----------------------------------------------------------------------------------
//                                              Exercise 1
//                                              ----------
//
// General : The program checks whether the required sum of two numbers
//           is in an ordered array.
//
// Input   : None.
//
// Process : Uses a function called "SumInArray".
//
// Output  : If the required sum of two numbers is found in an ordered array (BOOLEAN).
//
//-----------------------------------------------------------------------------------
// Programmer : Cohen Idan
// Student No : 211675038
// Date       : 23.10.2019
//-----------------------------------------------------------------------------------
void main(void)
{
    int sort_numbers[N] = {1,2,3,4,5};
    BOOLEAN answer = FALSE;
    answer = SumInArray(sort_numbers, N, S);

```

```
73      printf("Answer: %hu\n", answer);
74  }
```