Lab1 Sum of K

Problem Description

It is the same problem as in Exercise1, but you have to write a different program.

We are given an input file that contains

- word **SumOfK**,
- “target” number *K*,
- and a sequence of *N* numbers.

We want to determine if there are two numbers whose sum equals the given “target” number *K*.

For instance, if the input file contains

SumOfK	// word
10	// “target” number *K*
8 4 1 6	// sequence of *N* numbers

We know from the file that *K* is 10, sequence of numbers is 8 4 1 6, and number of elements *N* is 4 (we can count numbers). In this case, the answer is yes, there are two such numbers (4 and 6), because 4+6 is 10.

One number may be used twice (doubled). If the input file is
SumOfK
10
8 4 5 3

the answer is also yes, because 5+5 is 10.

Lab1 Program

Devise and implement an O(*N**log(*N*)) algorithm to solve the problem. Split it in two parts - sort the items first, after it is done, you have to search for numbers in O(*N*). See https://en.wikipedia.org/wiki/Heapsort

Code the solution. Read data from input file **in***X***.txt** and print the results to output file **out***X***.txt** (where *X* is 1, 2, …).

The differences between Lab1 and Exercise1 are

- The algorithm has to consist of two parts - O(*N**log(*N*)) sorting and O(*N*) searching
- The output file has also to contain the sorted list of input numbers

Input and Output Files

Test and Sample Files

There are five test input files provided – **in1.txt**, **in2.txt**, **in3.txt**, **in4.txt** and **in5.txt**. There are also four sample output files corresponding to the input files – **out1_sample.txt**, **out2_sample.txt** , **out3_sample.txt** and **out4_sample.txt** (file **out5_sample.txt** is not provided).

	Correspondent files	**Correspondent files**	**Correspondent files**	**Correspondent files**	**Correspondent files**
Provided test input files	in1.txt	in2.txt	in3.txt	in4.txt	In5.txt
Provided sample output files	out1_sample.txt	out2_sample.txt	out3_sample.txt	out4_sample.txt	**none**
Output files to produce	*out1.txt*	*out2.txt*	*out3.txt*	*out4.txt*	*out5.txt*

The format of the produced output files shall be similar to the provided sample output files. The results have to be the same.

Run your program using all five test input files to produce five output files and submit the input and output files together with the source code. Compare the output files produced by your program with the sample files. Submit the test input and the produced output files together with the program source code.

Examples

An example of two numbers

In the case of input file

```
SumOfK
10
8 4 1 6
```

Output file has to be

```
10
8 4 1 6           // original data
1 4 6 8           // sorted data
Yes
4+6=10
```

An example of one doubled number

In the case of input file

```
SumOfK
10
8 4 5 3
```

Output file has to be

```
10
8 4 5 3
3 4 5 8
Yes
5+5=10
```

An example where there is no solution

In the case of input file

```
SumOfK
11
8 4 5 1
```

Output file has to be

```
11
8 4 5 1
1 4 5 8
No
```