CSE 102 Homework Assignment 2 (Due: Oct 9 11:55 pm)

You are going to write a complete C program which implements the following functionality:

- The program reads real numbers from a file. Determines the chunks according to a criteria. For each chunk, the average of the numbers is calculated. Chunks are listed in ascending order based on their average.
- Input file contains a sequence of real numbers which are separated by whitespace. The whole sequence consists of chunks which are separated by three consecutive "zeros".(A separator)

-Each line of the output file lists the numbers in chunks.

- Code it so that it reads a text file named `input.txt` and writes to a text file named `output.txt`. (If you don't follow this convention your grade will be `0.0`)

Example

Contents of the input file:

```
12.432 23.5 344.6 11.85 0.0 0.0 0.0 2.5 8.2313 19.27 0.0 0.0 0.0 70.001 23.
64 13.62
```

Here, there are 3 chunks:

```
12.432 23.5 344.6 11.85
2.5 8.2313 19.27
70.001 23.64 13.62
```

Find averages of numbers in each chunk. Create a text file with chunks ordered as stacked lines based on the calculated average of each.

```
2.5 8.2313 19.27
13.62 23.64 70.001
11.85 12.432 23.5 344.6
```

Each line is a chunk. Row order is according to the average(the chunk with the smallest average is on the first row.)

Remarks:

- Maximum length of the input sequence is `1000`.
- Minimum length of a chunk is `1`.
- If there is more than 3 consecutive "zeros" (i.e. `0.0 0.0 0.0 0.0 ..`), the first three ones are considered a separator sequence and the rest are assumed to be included in the chunk following the separator sequence.
- Example:

```
<-------chunk--------><separator-><----chunk----------><separator-><----ch
unk-------->
12.432 23.5 344.6 11.85 0.0 0.0 0.0 0.0 2.5 8.2313 19.27 0.0 0.0 0.0 70.001
23.64 13.62
```

- There is at least one chunk in the sequence.
- Sequence starts with a chunk.
- If nothing follows the separator sequence, it is still a separator sequence and not included in any chunk.
- You don't have to do error checking on the input file. You can safely assume that you will be given a proper input file which doesn't violate the described format.

Turn in:

A complete C program `<assignment_2_name_id.c>` which can be compiled using the following command:

```
gcc -std=c99 assignment_2_name_id.c -o assignment_2_name_id
```

Caution:

- Read and apply "Assignment Submission Rules and Other Related Information" document which available on the class e-learning system.
- You may or may not get partial credit depending on how you structured or documented your code.