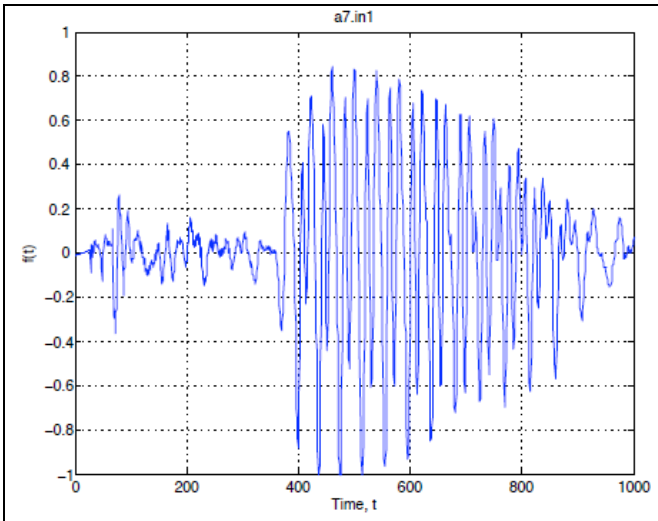
Purpose

The purpose of this assignment is to give you practice with single dimensional arrays as well as modularizing your program with more functions.

Scenario



An acoustical signal can be converted into an electrical signal by a microphone and the electrical signal can then be converted into a series of numbers representing the value of the electrical signal at discrete time intervals. These values have been stored in a data file. We are interested in analyzing this data to measure various aspects of the acoustical signal.

Problem

Read in a file containing a noise signal and print out several pieces of information on the signal.

Input

The input files consist of a series of a series of floating point numbers representing the acoustical signal. You should read these values into an array until end of file or until the array is full, use an array of data type double, of size 1500.

Output

The output should consist of a neat well labeled message with the following information in this order:

1. Number of sample points
2. The variance
3. The standard deviation
4. Average power
5. Average magnitude
6. Positive count
7. Number of zero crossings
8. Maximum change index

Details leading to computation of each output

1. The number of data points in the sample.
2. The **variance** (denoted σ^2) of the data points in the sample: the variance of a list of n values $x_0, x_1, x_2, \cdots, x_{n-1}$ is given by:

$$\sigma^2 = \frac{1}{n-1}\sum_{k=0}^{n-1}(x_k - \mu)^2$$

   where μ is the mean value ($\mu = \frac{1}{n}\sum_{k=0}^{n-1} x_k$).
3. The **standard deviation** of the data points in the sample: the standard deviation of a list of n values $x_0, x_1, x_2, \cdots, x_{n-1}$ is the square root of the variance.
4. The average power of the signal: the average power of a signal $x_0, x_1, x_2, \cdots x_{n-1}$ is defined as:

$$\frac{1}{n}\sum_{k=0}^{n-1} x_k^2$$

5. The average magnitude of the signal: the average magnitude of a signal $x_0, x_1, x_2, \cdots x_{n-1}$ is defined as:

$$\frac{1}{n}\sum_{k=0}^{n-1} |x_k|$$

6. The number of positive data points. The count of the number of data points that are greater than 0.
7. The number of zero crossings. The number of zero crossings is the number of times the signal changes from positive to negative or negative to positive. For this we will consider 0 as "positive", that is a change between a negative and 0 is a zero crossing while a change between a positive and 0 is not.
8. Index of maximum change. Find the index n for which $|x_n - x_{n+1}|$ is maximum. If there are less than two data points then return -1.

Other Details

- In this program you do not need to prompt for input or echo input. You should read the data from standard input as usual using scanf and use redirection to read the data from the input files.
- The array size (1500) must be defined as a constant. If there are more than 1500 input values (maximum size of the array) you should stop reading at 1500. Your program should handle empty files without crashing.
- You must use functions to modularize your solution, in particular you must write a function for each of the following:
    - Read the data into the array (function value returns actual size of the array)
    - Compute the variance
    - Compute the standard deviation
    - Compute both the average power and the average magnitude
    - Compute the positive count
    - Compute the number of zero crossings
    - Compute the maximum change index
- The first function (reading data into the array) must return the size of the array as a function value. The function that computes both the average power and average magnitude must return these two computed values as output arguments (function return value is void). Each of the rest of the functions must return the computed value as a function return value. While the first function takes a single argument which is the array, the rest of the functions take two input arguments, the first being the array and the second being the size of the array.
- Each function definition must have a boxed comment
- Sample input and output files are available.

Details

- You must follow all the coding style rules as specified in our "coding guidelines". In particular:
  - You must put your name enclosed in a comment box at the top, and keep any other comments that are already there.
  - Keep lines to the point of making your code easily readable.
  - You must use good names for any variables you create (a full word that describes what it is there for).
- Details that you do not follow are penalized after other scored items are added up, so even if you got a 100 for the functionality of your program, you can still get a lower score because you did not follow all the other requirements for the assignment.

Submission

Submit this assignment with the code 6P followed by the name of your source file – for example if `signal.c` is the name of your file:

```
submit 6P signal.c
```