Lab1 - Regular expressions

There is given a catalogue containing articles ending with *.html extension. Create a program, which gathers and returns following information about each file:

- filename
- author of the article
- number of sentences
- number of different abbreviations (i.e., words containing maximally three characters and ended with a point)
- number of different integer numbers belonging to range 0 do 65536
- number of different floating point numbers
- number of different dates
- number of different email addresses

Information about author and keywords can be obtained from section <META>.

Remaining information can be obtained from text between the first <P> tag (or its variant) and the first <META> tag.

It can be assumed, that each sentence finishes with point/dot (.), exclamation mark (!) or question mark (?), or a sequence of these characters, e.g. ??!. A sentence can also finish with a newline character (please, ignore possible html tags before newline character). We assume, that a point being a part of abbreviation, floating point number, date or email does not finish as sentence.

Regular expression for integer numbers should take into account specified range (please, do not use comparison operators, e.g. >, <, ==).

Regular expression for floating point numbers should accept, for example, the following numbers:
1.0, -1.0, 1., .5, 1.2e+3

Regular expression for dates should accept roughly correct dates. We can assume that each year contains 12 months and each month contains 31 days.
Dates can occur in following formats:
dd-mm-rrrr or *dd/mm/rrrr* or *dd.mm.rrrr* or in format
rrrr-dd-mm or *rrrr/dd/mm* or *rrrr.dd.mm*
Two dates are different if they represent different points in timeline . Two dates representing the same year, month and day are the same even if they are in different format.

In email address, part following @ character must contain at least one point. This part of email cannot start of finish with a point. Two consecutive points are also forbidden in an email address.

The program execution should be invoked as follows:

```
python lastname.py catalogue_name
```

Results should be printed to standard output.
Program should be implemented in Python 3.
Please, In comments do not use diacritic characters
Please, to produce indentation, use space characters, do not use tab characters.