"""
1. A hapax legomenon (often abbreviated to hapax) is a word which occurs only once in either the written record of a language, the works of an author, or in a single text. Define a function that given the file name of a text will return all its hapaxes. Make sure your program ignores capitalization.
"""

```
import os
import string

def replaceWord(stringValue):
    dataPunctuation = string.punctuation

    for i in dataPunctuation:
        stringValue = stringValue.replace(i, '')

    return stringValue

def hapax(filename):
    filename = filename.lower()

    readFile = open(os.getcwd() + '\\'+filename, 'r')
    data = readFile.read().lower()
    data = data.split()

    listOccurs = {}
    for i in data:
        current = replaceWord(i)
        listOccurs.setdefault(current, 0)
        listOccurs[current] = listOccurs[current] + 1

    print("Total Words : " + str(len(listOccurs)))
    print("--------------------------------------------------------------------------------")
    print("Hapax word : ")
    print("--------------------------------------------------------------------------------")
    for i in listOccurs.keys():
        nOccurs = listOccurs.get(i)
        if(nOccurs == 1):
            print(i , end=",")

    print()
    print("Most common : ")
    print("--------------------------------------------------------------------------------")
    maximum = max(listOccurs, key=listOccurs.get)
    print(maximum, listOccurs[maximum])

hapax("books.txt")
```

"""
2. Write a program that given a text file will create a new text file in which all the lines
from the original file are numbered from 1 to n (where n is the number of lines in the file).
"""

```
import os

def createNewFile(filename):
    newPath = os.getcwd() + "\\newFile.txt"
    filename = filename.lower()
    readFile = open(os.getcwd() + "\\" + filename, 'r')

    newFile = open(newPath, 'w')

    n = 1
    for i in readFile:
        newFile.write(str(n) + ". " + i)
        n = n + 1

    readFile.close()
    newFile.close()
    print("New File Created\n" + "Location : "+newPath)

createNewFile("originalFile.txt")
```

"""
3. Write a program that will calculate the average word length of a text stored in a file
(i.e the sum of all the lengths of the word tokens in the text, divided by the number of word tokens).
"""

```
import os

def getAverageWord(filename):
    readFile = open(os.getcwd() + "\\" + filename, 'r')
    data = readFile.read().split()

    numberOfWord = len(data)
    totalWordLength = len(''.join(data))

    return totalWordLength / numberOfWord


print("Average word length is "+str(getAverageWord("books.txt")))
```

"""
4. A sentence splitter is a program capable of splitting a text into sentences.
The standard set of heuristics for sentence splitting includes (but isn't limited to) the following rules:
Sentence boundaries occur at one of "." (periods), "?" or "!", except that .
Periods followed by whitespace followed by a lower case letter are not sentence boundaries.
a. Periods followed by a digit with no intervening whitespace are not sentence boundaries.
b. Periods followed by whitespace and then an upper case letter,
but preceded by any of a short list of titles are not sentence boundaries.
Sample titles include Mr., Mrs., Dr., and so on.
c. Periods internal to a sequence of letters with no adjacent whitespace are not sentence boundaries
(for example, www.aptex.com, or e.g).
d. Periods followed by certain kinds of punctuation (notably comma and more periods) are probably not sentence boundaries.
Your task here is to write a program that given the name of a text file can write its
content with each sentence on a separate line. Test your program with the following short text:
Mr. Miyagi bought cheapsite.com for 1.5 million dollars, i.e. he paid a lot for it.
Did he mind? Adam Jones Jr. thinks he didn't. In any case, this isn't true...
Well, with a probability of .9 it isn't. The result should be:
Mr. Miyagi bought cheapsite.com for 1.5 million dollars, i.e. he paid a lot for it.
Did he mind?
Adam Jones Jr. thinks he didn't.
In any case, this isn't true...
Well, with a probability of .9 it isn't.
"""

```
import os

def convertToSentence(filename):
    readFile = open(os.getcwd() + "\\" + filename, 'r')
    stringValue = readFile.read()

    titles = ["Mr.", "Mrs.", "Dr."]
    data = stringValue.split()

    result = ""
    n = 0
    for i in data:
        temp = i + " "
        if(i.endswith(".") or i.endswith("?") or i.endswith("!")):
            if(n != len(data) -1):
                if(i not in titles and str(data[n + 1][0]).isupper()):
                    temp = temp + "\n"

        n =  n + 1

        result = result + temp

    return result

print(convertToSentence("shortText.txt"))
```