CSCI 235, Programming Languages, Python 1

Deadline: 05/07.11.2018 (the day of your lab)

Goal of this exercise is that you understand how to run a Python program, and that you gets some familiarity with the `scipy` library.

- **Read the complete task before starting!**
- **Submit the final answer into Moodle as a single file myname.py**
- **Answers that show evidence of lazyness, or are incomplete may get rejected!**

Start the **python** interpreter. Make sure that you have version 3 or higher. If at some point, you wish to leave the interpreter, type `exit( )` In the interpreter, type `from scipy import *` and `from scipy.linalg import *`

1. We will study linear algebra in `scipy`. The matrices are similar to those in C^{++} task 2. Type

   ```
   m1 = array( [ [ 1/2, 1/3 ], [ -2/7, 2/ 8 ]] )
   m2 = array( [ [ -1/3, 2/7 ], [ 2/5, -1/7 ]] )
   m3 = array( [ [ -1/5,2/3], [1/8, 3/11] ] )

   p = array( [ [ -1/30, 2/21 ], [ 41/210, -23/196 ] ] )

   I = array( [ [ 42, -56 ], [ 48, 84 ]] ) / 37
   ```

   We remember from our C^{++} exercises that $p = m_1 \times m_2$. If you type `m1*m2`, the interpreter will compute the product memberwise, which is not what we want.

   You can type `dot(m1,m2)` and compare the result to `p`, for example by typing `dot(m1,m2) - p`. Using `==` does not work, because of floating point inaccurracy.

2. Now we also want to verify that the inverse of m_1 equals I. The function is called `inv`. One can also apply it on m_1, and verify that the inverse of m_1 equals I by subtracting the results.

3. Verify that matrix multiplication is associative,

$$(m_1.m_2).m_3 = m_1.(m_2.m_3).$$

4. Verify that matrix multiplication is distributive on both sides:

$$\begin{cases} m_1.(m_2 + m_3) = m_1.m_2 + m_1.m_3 \\ (m_1 + m_2).m_3 = m_1.m_3 + m_2.m_3 \end{cases}$$

5. Verify that matrix multiplication corresponds to composition of application:

$$m_1(m_2(v)) = (m_1.m_2)(v)$$

A vector can be created by typing `v = array( [ 3.0, -1 ] )`. Application of a matrix on a vector is also done by the function `dot`.

6. Determinant commutes over multiplication:

$$\det(m_1).\det(m_2) = \det(m_1.m_2).$$

The determinant function is called `det`.

7. Inverse of matrix is indeed inverse:

$$m.\mathrm{inv}(m) = \mathrm{inv}(m).m = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

8. Finally, prepare a file **yourname.py** and put your answers in this file, in the following form:

```
from scipy import *

def solution( ) :
   print( "here are the answers of John Cleese" )

   m1 = array( [ [ 1/2, 1/3 ], [ -2/7, 2/ 8 ]] )
   m2 = array( [ [ -1/3, 2/7 ], [ 2/5, -1/7 ]] )
   m3 = array( [ [ -1/5,2/3], [1/8, 3/11] ] )

   print( "This is part 1\n" )

   print( "the product of m1 and m2 is " )
   print( dot(m1,m2) )
```

You can run the file by typing `import yourname`, and `yourname.solution( )`

9. Submit this file into Moodle. Note that any solutions that show evidence of lazyness (including solutions that cause errors) will get rejected.