NPRG065: PROGAMMING IN PYTHON PRACTICALS 9



MATEMATICKO-FYZIKÁLNÍ FAKULTA Univerzita Karlova





1. Implement a Rectangle class representing a rectangle

```
    Usage:
    r = Rectangle(4, 5)
    print(r.get_area()) # prints out 20
    r.set_size(2, 6)
    print(r.get_area()) # prints out 12
```



- 2. Implement a Matrix class representing mathematical matrix
 - Try to provide implementation for sparse matrices
 - Implement methods for basic operations over matrices (sum, product,...)
 - The methods should return new instances of Matrix



3. Implement a simple logging library

```
    class Logger:
__init__(self, name)
def set_level(self, level)
def log(self, level, message)
def add_printer(self, printer)
```

- The log method prints out messages only if the given level is higher than the level set via set_level()
- Each message is printed using each of the added printers
- A printer is an object containing the print(message) method
- Create several printers, printing to std out, to std err, to file



4. Extend the printer to contain a formatter

- A formatter is an object with the format(logger, message) method, which accepts a message and returns the message prepared for printing
- Implement several formatters, e.g., with the name of the logger, with the current data and time,...
- Current date/time:

import datetime
print(datetime.datetime.now())



The slides are licensed under Creative Commons Attribution-NonCommercial 4.0 International License

https://creativecommons.org/licenses/by-nc/4.0

