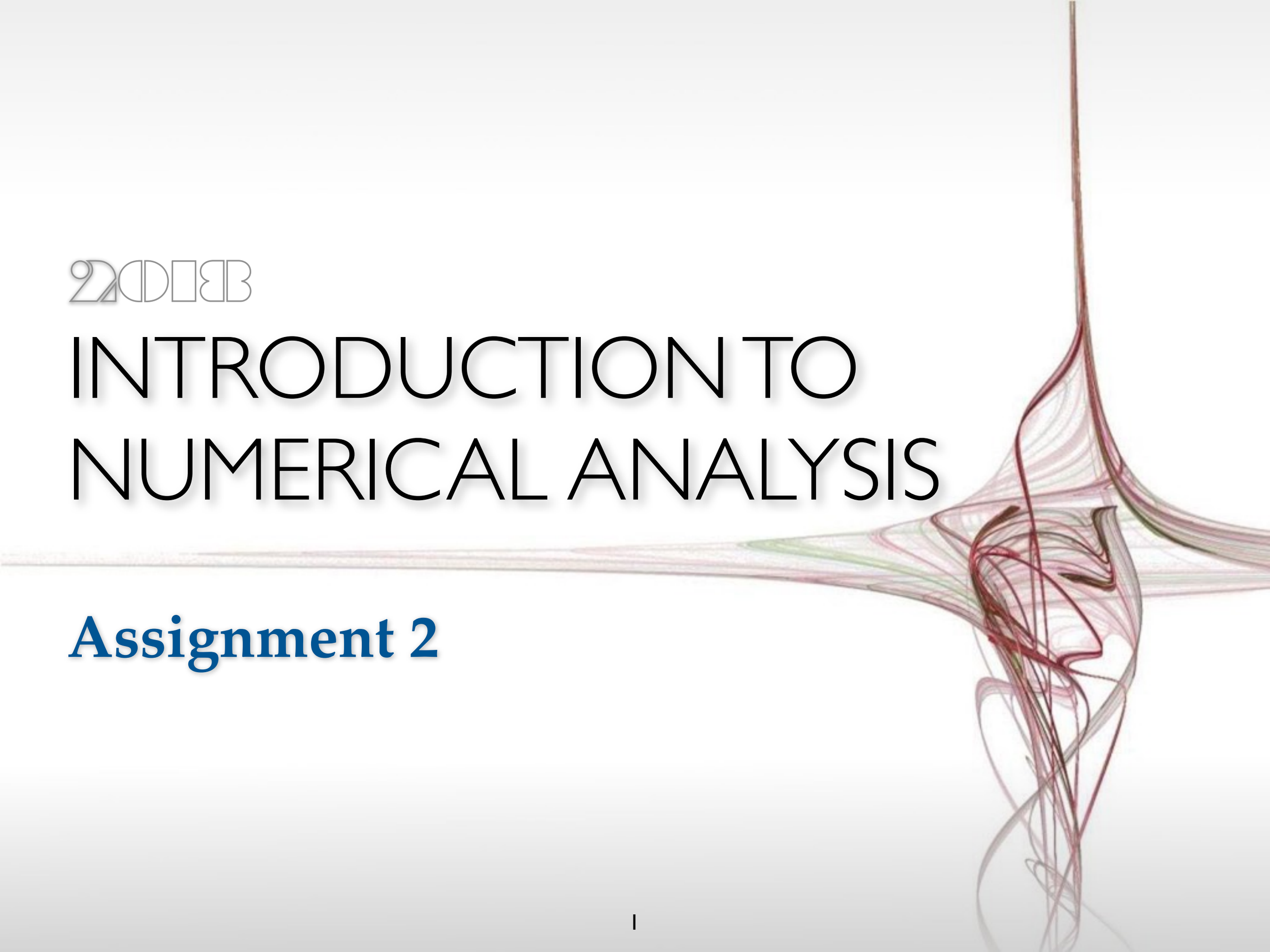
2018

INTRODUCTION TO NUMERICAL ANALYSIS



Assignment 2

REMINDER

- **Please download the template code from CEIBA and implement your answer accordingly!**
- You will have **maximum 2 weeks** to answer the assignment, you can discuss with our people or googling, but please do not just COPY the answers!
- Remember: you have to upload <u>your code</u> (as a .py file), not the snapshot of your output screen!
- Mini-TA should check whether your code is working properly or not. If not working properly, mini-TA will ask you to correct it.

REMINDER (II)

- **Please download the template code from CEIBA and implement your answer accordingly!**

陳凱風 老師，歡迎進入 Ceiba 系統

課程資訊
教師資訊
課程內容
討論看板
作業區
資源分享
投票區
登出

語言 Language:
中文(Chinese)

更新：2018-03-18
訪客：336

作業區

作業列表

已指派的作業

名稱	成員	繳交方法	成績比重	繳交期限	逾期繳交	繳交日期	作業評語	作業觀摩
Assignment 1-1	個人	線上繳交	--	2018-03-26 17	可以	--	--	進入
Assignment 1-2	個人	線上繳交	--	2018-03-26 17	可以	--	--	進入
Assignment 1-3	個人	線上繳交	--					

線上繳交

名稱	Assignment 1-1
作業說明	
相關檔案	檔案
相關網址	
成員	個人
繳交方法	線上繳交

The template code is here!

ASSIGNMENT 2-1

Give me a NumPy array #1

- Construct a function which takes an **input integer n**, and output a two-dimensional **10×10** NumPy array with the following pattern:

```
array([[0, n, 0, 0, 0, 0, 0, 0, n, 0],
[n, 0, n, 0, 0, 0, 0, n, 0, n],
[0, n, 0, n, 0, 0, n, 0, n, 0],
[0, 0, n, 0, n, n, 0, n, 0, 0],
[0, 0, 0, n, n, n, n, 0, 0, 0],
[0, 0, 0, n, n, n, n, 0, 0, 0],
[0, 0, n, 0, n, n, 0, n, 0, 0],
[0, n, 0, n, 0, 0, n, 0, n, 0],
[n, 0, n, 0, 0, 0, 0, n, 0, n],
[0, n, 0, 0, 0, 0, 0, 0, n, 0]])
```

where **'n'** above should be replaced with the input integer.

ASSIGNMENT 2-2

Give me a NumPy array #2

- Construct a function which takes an **input integer n**, and output a two-dimensional **n×n** NumPy array with the following pattern:

```
array([[ 1,  2,  3,  4,  5, ..., n   ],
       [ 2,  3,  4,  5,  6, ..., n+1 ],
       [ 3,  4,  5,  6,  7, ..., n+2 ],
       [ 4,  5,  6,  7,  8, ..., n+3 ],
       [ 5,  6,  7,  8,  9, ..., n+4 ],
       [ ...                   , ..., ... ],
       [ n,n+1,n+2,n+3,n+4, ..., 2n-1]])
```

ASSIGNMENT 2-3

Game of Life

- The “Game of Life” is a cellular automaton devised by John Horton Conway in 1970.
- Basically it is a zero-player game, which takes an initial state and simply evolves by itself. No further input is required.
- It starts with an infinite two-dimensional orthogonal grid of square cells, each of which is in one of two possible states, **alive** or **dead**.
- Every cell interacts with its **8 neighbours**. At each step in time, its state depends on **the number of its alive neighbours**.

ASSIGNMENT 2-3 (CONT.)

- The Game of Life rules:
  - Any live cell with fewer than **2** live neighbours dies, as if caused by underpopulation.
  - Any live cell with **2** or **3** live neighbours lives on to the next generation.
  - Any live cell with more than **3** live neighbours dies, as if by overpopulation.
  - Any dead cell with exactly **3** live neighbours becomes a live cell, as if by reproduction.
- Assignment: implement the rules above and **print out the evaluated pattern of this "I-Column" for 15 iterations**. Take the template code please!