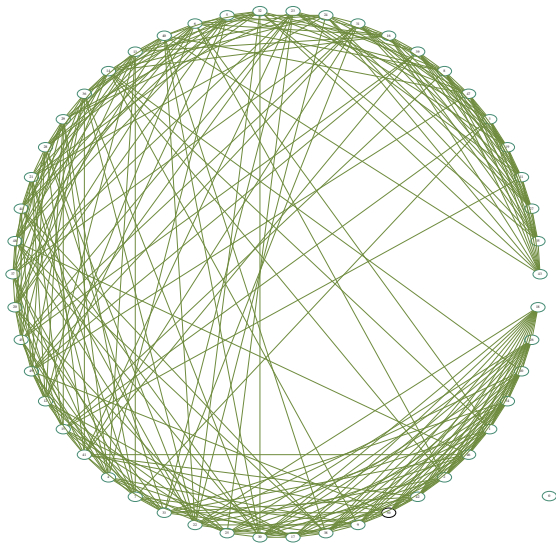
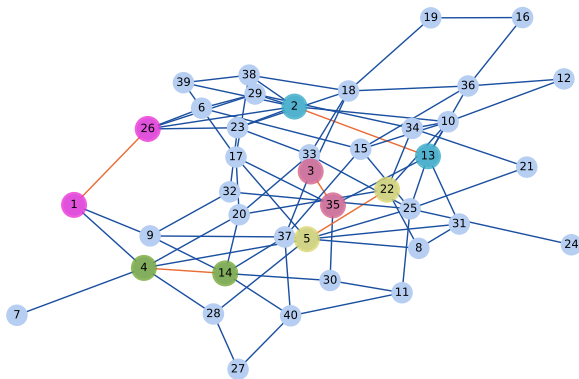


# Algorithmic complexity and graphs: overview of the module

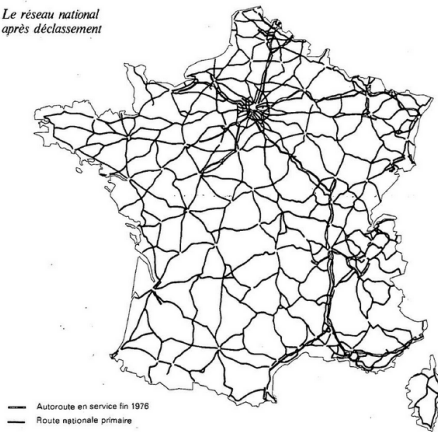
27 septembre 2023



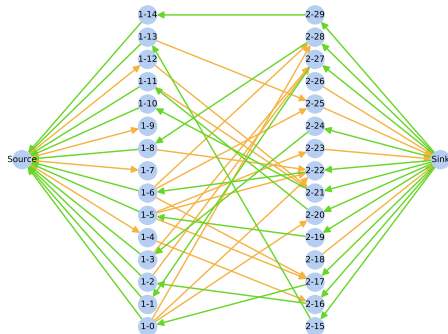
Matching size: 5  
Algo step: 19  
Nb nodes: 40

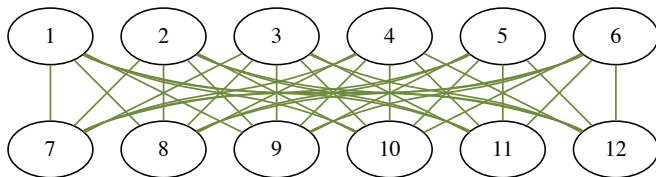


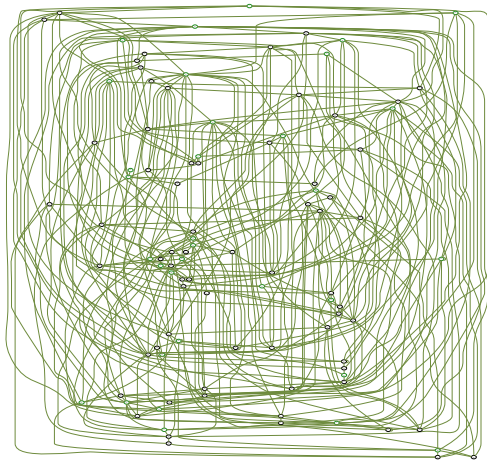
*Le réseau national  
après déclassement*



residual graph step 12

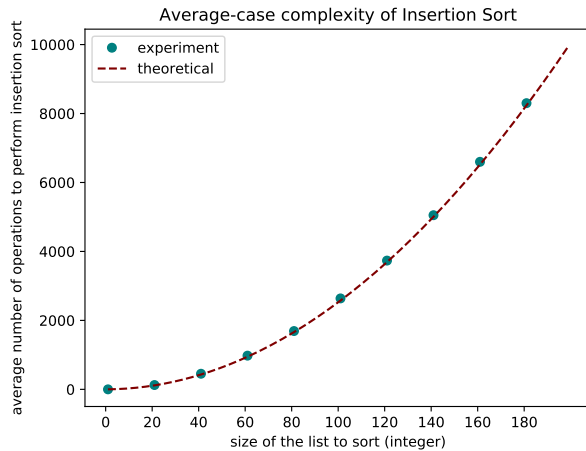






```
→ rsa git:(correction) X pp cipher_rsa.py
public key : (187, 153)
private key : 137
keys are ok : b is the inverse of a modulo phi
cipher rsa
a (97) becomes 124
l (108) becomes 113
g (103) becomes 86
o (111) becomes 111
r (114) becomes 141
i (105) becomes 150
t (116) becomes 139
h (104) becomes 70
m (109) becomes 10
(32) becomes 32
c (99) becomes 88
o (111) becomes 111
u (117) becomes 134
r (114) becomes 141
s (115) becomes 81
e (101) becomes 118
code : 124,113,86,111,141,150,139,70,10,32,88,111,134,141,81,118
```





```
for i in range(n):  
    for j in range(i):  
        l = [i+j+k for k in range(n)]
```

# Overview of the module

- Day 1 Reminders on algorithms, cryptography, recursion, knapsack, shortest path
- Day 2 Computation of complexities, graph problems (dominating set and others), notion of classes of complexity
- Day 3 Networks, the matching problem and the maximum flow problem

# Organization

- ▶ Presentations / discussions
- ▶ Coding exercises / paper+pen exercises
- ▶ Project : explained saturday
- ▶ **questions** : please feel free to ask questions :
  - ▶ you can ask directly
  - ▶ or write them in the chat

more questions = more interesting / fun course !

# Organization

- ▶ Please clone the following repository :  
`https://github.com/nlehir/AlgoGraph.git`, that contains :
  - ▶ slides
  - ▶ exercises
  - ▶ other useful information

## Practical aspects

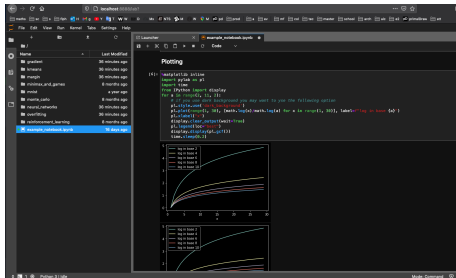
- ▶ Python 3 <https://www.python.org/>
- ▶ Third-party libraries : see **requirements.txt**
- ▶ For installation, several options are available :
  - ▶ create a folder for the course and install libraries in a virtual environment (using e.g. pip)  
<https://docs.python.org/3/tutorial/venv.html>
  - ▶ install libraries globally on your machine, using e.g. pip (not recommended in the python community for production projects)

# Virtual environment

```
..aph/AlgoGraph (-zsh)
(env) → AlgoGraph git:(master) ✕ which python
/Users/nico/Desktop/enseignement/epitech/AlgoGraph/AlgoGraph/env/bin/python
(env) → AlgoGraph git:(master) ✕ python --version
Python 3.9.0
(env) → AlgoGraph git:(master) ✕ pip list
Package Version
-----
anyio 3.6.1
appnope 0.1.3
argon2-cffi 21.3.0
argon2-cffi-bindings 21.2.0
asttokens 2.0.8
attrs 22.1.0
Babel 2.10.3
backcall 0.2.0
beautifulsoup4 4.11.1
bleach 5.0.1
certifi 2022.9.24
cffi 1.15.1
charset-normalizer 2.1.1
contourpy 1.0.5
cyclcr 0.11.0
debugpy 1.6.3
decorator 5.1.1
defusedxml 0.7.1
entrypoints 0.4
executing 1.1.0
```

# Organization

- ▶ **jupyter notebooks** are convenient python interpreters. Depending on your preference, you may use them or mere python scripts (I tend to prefer scripts but by copying and pasting, you are able to turn a script into a notebook and vice versa). Vs code supports notebooks.





## Contact

firstname lehir @ gmail.com