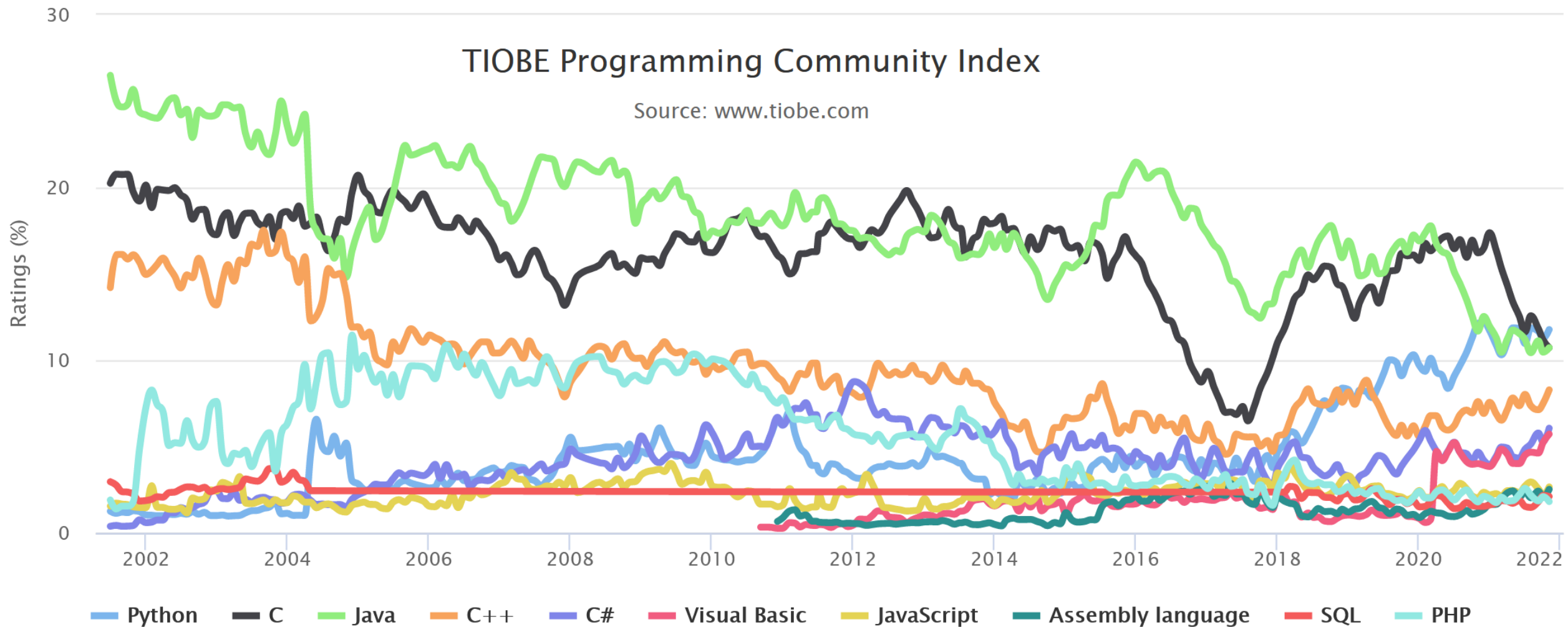


# "Python med Gerth"

*Historier fra kurset*  
*Introduktion til Programmering med Videnskabelige Anvendelser*



- Siden november 2021 – Python mest populære programmeringsprog (ifølge TIOBE)

# Er Python et langsomt programmeringssprog?

n	C (gcc 9.2)	C++, int (g++ 9.2 )	C++, long (g++ 9.2 )	Java (12.0)	Python		
					Python (3.8.1)	PyPy (7.3.0)	Numba, int64
$10^7$	0.001 sec*	0.001 sec*	0.003 sec	0.006 sec*	1.5 sec	0.27 sec	0.002 sec
$10^9$	0.10 sec*	0.10 sec*	0.30 sec	0.40 sec*	145 sec	27 sec	0.2 sec

\* Forkert output (overløb, 32 bit integer)

- Beregning af  $1 + 2 + 3 + \dots + n$

# Datalogi Studierevision 2003

1. semester (efterår)	1. kvarter	Introduktion til programmering	Perspektiverende datalogi	Calculus 1
	2. kvarter	Programmering 2	Computerarkitektur	Calculus 2
2. semester (forår)	3. kvarter	Algoritmer og datastrukturer 1	Webteknologi	Interaktionsdesign
	4. kvarter	Algoritmer og datastrukturer 2	Programmeringssprog	Regularitet og automater

- Alle kurser 5 ECTS = 7 uger (tidligere 10 ECTS og 15 uger)
- 12 eksamener på 1. år
- "Introduktion til programmering" (5 ECTS) obligatorisk støttefag på matematik

# Datalogi Studierevision 2017

1. semester (efterår)	Introduktion til programmering	Algoritmer og datastrukturer	Calculus $\beta$
2. semester (forår)	Introduktion til databaser	Programmeringssprog	Lineær algebra
	Implementering og anvendelser af databaser		

- Matematik ønskede *ikke* "Introduktion til programmering" som støttefag
- Ønskede 5 ECTS Python kursus + 5 ECTS Projekt kursus → 10 ECTS Python
- Matematik-Økonomi ønskede C++ → Python

# Introduktion til Programmering med Videnskabelige Anvendelser (IPSA, 10 ECTS, foråret)

- Obligatorisk for
  - Matematik (4. semester)
  - Matematik-Økonomi (4. semester)
  - Tilvalg i Matematik (6. semester)
  - Datavidenskab (2. semester, siden 2020)
- Valgfag
  - Kemi (6. semester)
  - Cognitive science, fysik, ...
- Datalogi og IT-produktudvikling




	2018	2019	2020	2021	2022
Hold	4	6	7	8	10
Studerende	91	143	145	166	196

# IPSA 2022 - studiebaggrund

	196
Datavidenskab (bach)	27
Dramaturgi	1
Engelsk (AR)	1
Filosofi	1
Fysik	21
Historie	1
Idrætsvidenskab (Bach) (2019-)	2
Kemi	15
Kemiteknologi	4
Kinastudier	1
Kognitionsvidenskab (BSc)	6
Lingvistik	1
Matematik (NS)	43
Matematik-økonomi	55
Medicinalkemi	1
Musikvidenskab	3
Nanoscience	11
Samfundsfag	2

# IPSA starten

- Indhold defineret foråret 2017
- Udbudt siden foråret 2018
- Efteråret 2017 – en IPSA underviser siger op
- November 2017 – Gerth bliver kursusansvarlig
  - aldrig brugt Python
  - 2 måneder til kursusstart
  - planlagt lærebog ikke særlig anvendelig
  - *panik!*
- Python 2 eller Python 3?   $7 / 3 = 2$  eller  $7 / 3 = 2.333\dots$



# Kursus indhold

Basal programming  
Avanceret / specifik python  
Biblioteker & anvendelser

1. Introduction to Python	10. Functions as objects	19. Linear programming
2. Python basics / if	11. Object oriented programming	20. Generators, iterators, with
3. Basic operations	12. Class hierarchies	21. Modules and packages
4. Lists / while / for	13. Exceptions and files	22. Working with text
5. Tuples / comprehensions	14. Doc, testing, debugging	23. Relational data
6. Dictionaries and sets	15. Decorators	24. Clustering
7. Functions	16. Dynamic programming	25. Graphical user interfaces (GUI)
8. Recursion	17. Visualization and optimization	26. Java vs Python
9. Recursion and Iteration	18. Multi-dimensional data	27. Final lecture

**10 afleveringer**

**1 afsluttende projekt (sidste måned)**

# Opgave 4.5\*\* (sliced ranges)

## Python shell

```
> for i in range(10):
    print(i, end=' ')
| 0 1 2 3 4 5 6 7 8 9
> print(*range(10))
| 0 1 2 3 4 5 6 7 8 9
> print(*range(3, 100, 17))
| 3 20 37 54 71 88
> print(*range(3, 100, 17)[1:10:2])
| 20 54 88
```

- `range(x, y, z)[i:j:k]` returner en ny `range(a, b, c)`
- Lav et program der beregner `a`, `b` og `c`, givet `x`, `y`, `z`, `i`, `j` og `k`

# Dynamisk programmering

```
binomial.py
```

```
def binomial(n, k):  
    return binomial(n - 1, k - 1) + binomial(n - 1, k) if 0 < k < n else 1
```

```
Python shell
```

```
> binomial(10, 5)  
| 252  
> binomial(100, 50)  
| ...vil tage et par hundrede milliarder år
```

```
binomial.py
```

```
from functools import cache as dynamic_programming
```

```
@dynamic_programming
```

```
def binomial(n, k):  
    return binomial(n - 1, k - 1) + binomial(n - 1, k) if 0 < k < n else 1
```

```
Python shell
```

```
> binomial(100, 50)  
| 100891344545564193334812497256
```

```
def dynamic_programming(f):  
    answers = {}  
    def wrapper(*args):  
        if not args in answers:  
            answers[args] = f(*args)  
        return answers[args]  
    return wrapper  
  
binomial = dynamic_programming(binomial)
```

# List comprehensions

*Nogle instruktører mener at alle opgaver skal laves med list comprehensions*

## Python shell

```
> L = [1, 2, 3]
> squares = [x ** 2 for x in L]
> print(squares)
| [1, 4, 9]
```

## Python shell

```
> def permutation(L):
    return [[] if not L else [*P[:i], L[0], *P[i:]]
           for P in permutation(L[1:])]
           for i in range(len(P) + 1)]

> print(permutation(L))
| [[1, 2, 3], [2, 1, 3], [2, 3, 1], [1, 3,
```

```
def permutation(L):
    return [resList[-1]
           for resList in [ [[]]]
           if not [None]
           for elm in reversed(L)
           for Ps in [ [*P[:i], elm, *P[i:]]
                      for P in resList[-1]
                      for i in range(len(P) + 1) ] ]
           if resList.append(Ps)]

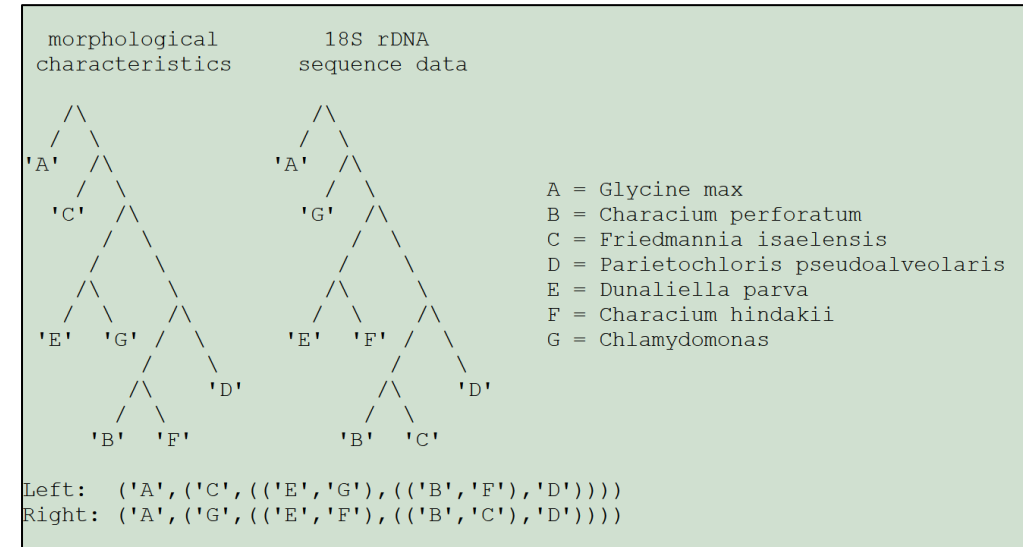
] [0]
```



# Afleveringer 3 og 4

## Triplet-afstanden

Gerth Stølting Brodal, Rolf Fagerberg, Thomas Mailund, Christian N. S. Pedersen, Andreas Sand: *Efficient algorithms for computing the triplet and quartet distance between trees of arbitrary degree*. SODA 2013: 1814-1832.



### triplet\_distance\_compact.py

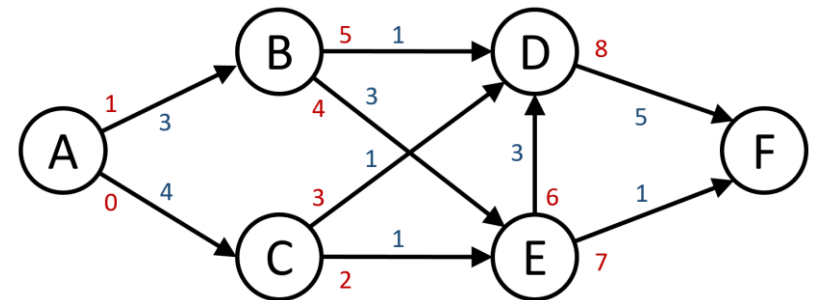
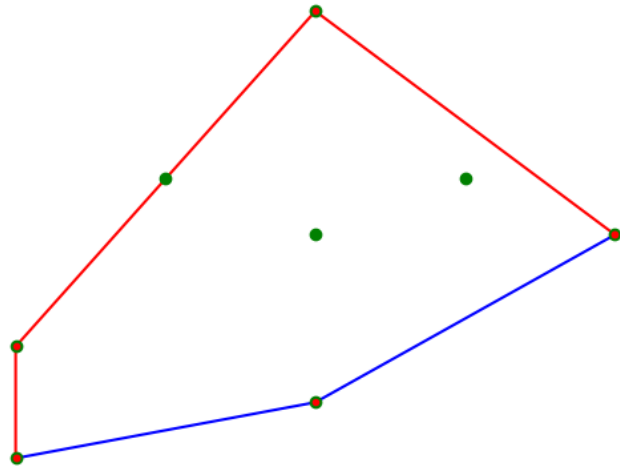
```
def triplet_distance(tree1, tree2):
    def compute(tree):
        if not isinstance(tree, tuple): return [tree], []
        (l1, lt), (r1, rt) = [compute(c) for c in tree]
        t = [(x, (y, z)) for a, b in [(l1, r1), (r1, l1)] for y in b for z in b if y < z for x in a]
        return l1 + r1, lt + rt + t

    (L1, T1), (L2, T2) = compute(tree1), compute(tree2)
    n = len(L1)
    return n * (n-1) * (n-2) // 6 - len(set(T1) & set(T2))
```

# Andre afleveringsopgaver

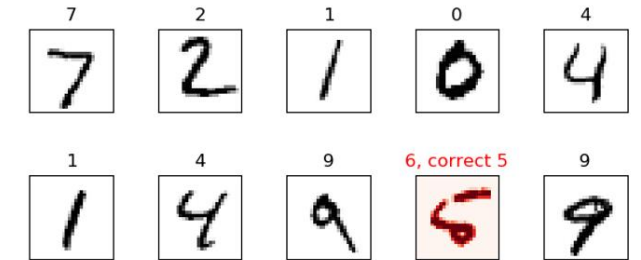
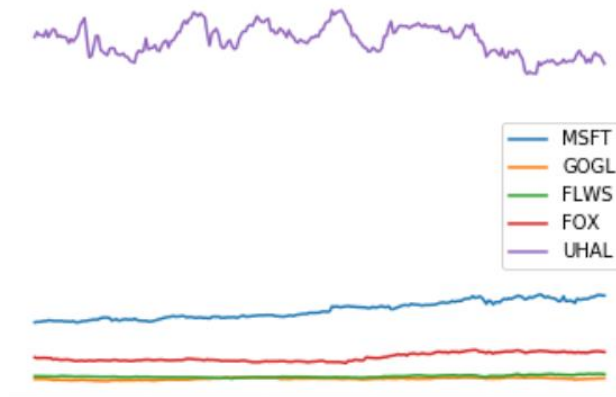
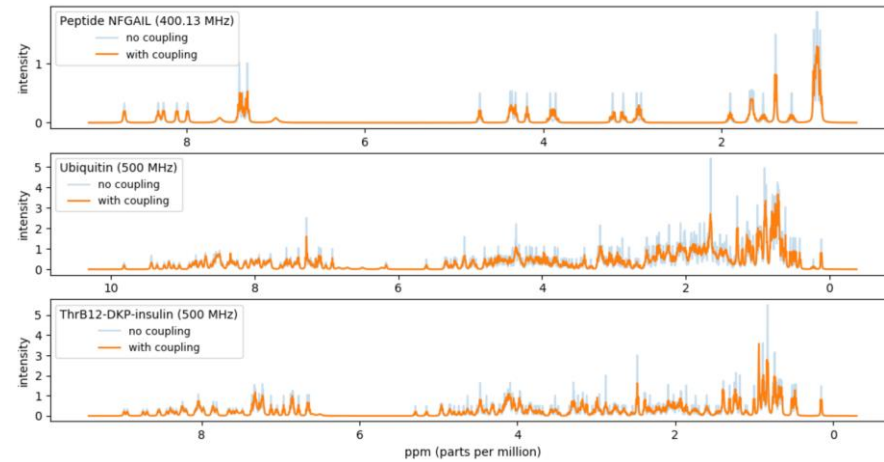
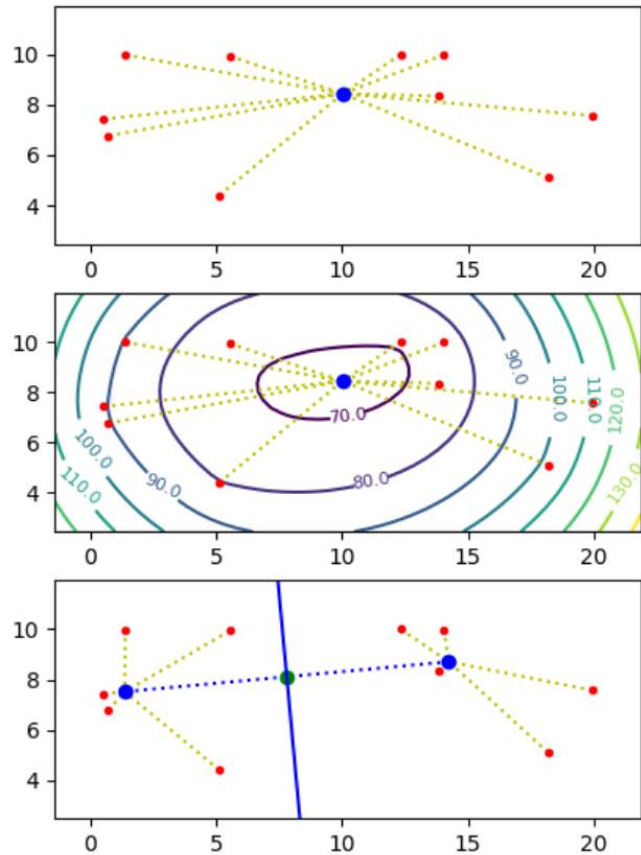
- 8 Queens
- Convex hull
- Maximum flow (vha. lineær programmering)

```
Size of board: 5
Q.....
..Q..
.....Q
.Q...
...Q.
```

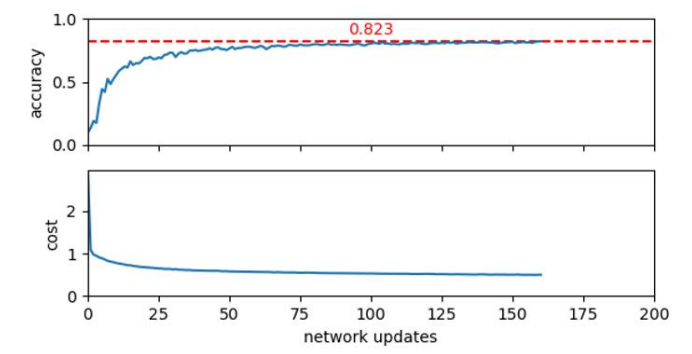
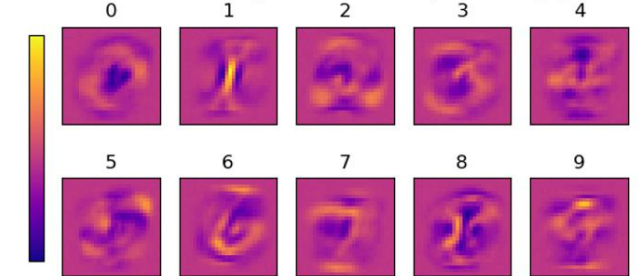


# Afsluttende projekter

Geometric median



Linear classifier weights  
(columns of weight matrix A reshaped to image)



# Eksamen før Covid19



## Question 1 (Expressions, 4 %)

What is the result of each of the below expressions ?

*Hvad er resultatet af hvert af nedenstående udtryk ?*

	0	0.0	1	1.0	ZeroDivisionError
0 / 1	<input type="checkbox"/> A	<input type="checkbox"/> B	<input type="checkbox"/> C	<input type="checkbox"/> D	<input type="checkbox"/> E
1 / 0	<input type="checkbox"/> A	<input type="checkbox"/> B	<input type="checkbox"/> C	<input type="checkbox"/> D	<input type="checkbox"/> E
0 // 1	<input type="checkbox"/> A	<input type="checkbox"/> B	<input type="checkbox"/> C	<input type="checkbox"/> D	<input type="checkbox"/> E
1 // 0	<input type="checkbox"/> A	<input type="checkbox"/> B	<input type="checkbox"/> C	<input type="checkbox"/> D	<input type="checkbox"/> E
1 # 0	<input type="checkbox"/> A	<input type="checkbox"/> B	<input type="checkbox"/> C	<input type="checkbox"/> D	<input type="checkbox"/> E
0 # 1	<input type="checkbox"/> A	<input type="checkbox"/> B	<input type="checkbox"/> C	<input type="checkbox"/> D	<input type="checkbox"/> E
1 % 0	<input type="checkbox"/> A	<input type="checkbox"/> B	<input type="checkbox"/> C	<input type="checkbox"/> D	<input type="checkbox"/> E
0 % 1	<input type="checkbox"/> A	<input type="checkbox"/> B	<input type="checkbox"/> C	<input type="checkbox"/> D	<input type="checkbox"/> E

## Question 7 (Lists, 4 %)

---

```
A = [[1, 2], [3, 4]]
```

```
B = A
```

```
B[1] = 5
```

```
C = A[:]
```

```
C[1] = 6
```

```
A[0][0] = 7
```

---

What is the value of **A**, **B** and **C** after executing the above code ?

*Hvad er værdien af **A**, **B** og **C** efter udførelsen af ovenstående kode ?*

	[[7, 2], [3, 4]]	[[1, 2], 5]	[[7, 2], 5]	[[1, 2], 6]	[[7, 2], 6]
A	<input type="checkbox"/> A	<input type="checkbox"/> B	<input type="checkbox"/> C	<input type="checkbox"/> D	<input type="checkbox"/> E
B	<input type="checkbox"/> A	<input type="checkbox"/> B	<input type="checkbox"/> C	<input type="checkbox"/> D	<input type="checkbox"/> E
C	<input type="checkbox"/> A	<input type="checkbox"/> B	<input type="checkbox"/> C	<input type="checkbox"/> D	<input type="checkbox"/> E

## Question 8 (List comprehension, 4 %)

---

```
print([x + x for x in [[1, 2], [3, 4]]])
```

---

What does the above code print ?

*Hvad udskriver ovenstående kode ?*

[4, 6]  A

[3, 6]  B

[2, 4, 6, 8]  C

[1, 2, 1, 2, 3, 4, 3, 4]  D

[[1, 2, 1, 2], [3, 4, 3, 4]]  E

[[1, 2, 3, 4], [1, 2, 3, 4]]  F

Covid 19

# Omlægning til online undervisning

- 11/3 2020 20:00 Statsministeren lukker Danmark ned
- **11/3 2020 23:00 Instruktorerne foreslår Discord**
- 12/3 2020 00:30 Discord server klar
- 12/3 2020 09:00 Første TØ på Discord
- 13/3 2020 10:00 Første studiecafé på Discord
- Forelæsninger lægges ud som YouTube videoer
- Programmeringseksamen med alle hjælpemidler, incl. internet

# Ny eksamensform 2020–

- Download og upload eksamen som .zip på eksamen.au.dk
- For hver opgave udleveres 5 test input og output
- Test script run\_tests.py til at køre tests under eksamen og identificerer fejl i output

```
tests
├── _README_.txt
├── A.py
├── B.py
├── C.py
├── D.py
├── E.py
├── F.py
├── G.py
├── H.py
├── I.py
├── J.py
├── K.py
├── L.py
└── run_tests.py
```

## Tests passed:

```
A.py 0/5
B.py 0/5
C.py 0/5
D.py 0/5
E.py 0/5
F.py 0/5
G.py 0/5
H.py 0/5
I.py 0/5
J.py 0/5
K.py 0/5
L.py 0/5
```

```
-----
Total 0/60
-----
```

Press [Enter] to exit

## A.py tests/A/A.in1 [failed]

```
Input
> World
Correct output
> Hello World.
Received output
*> Hello World
```

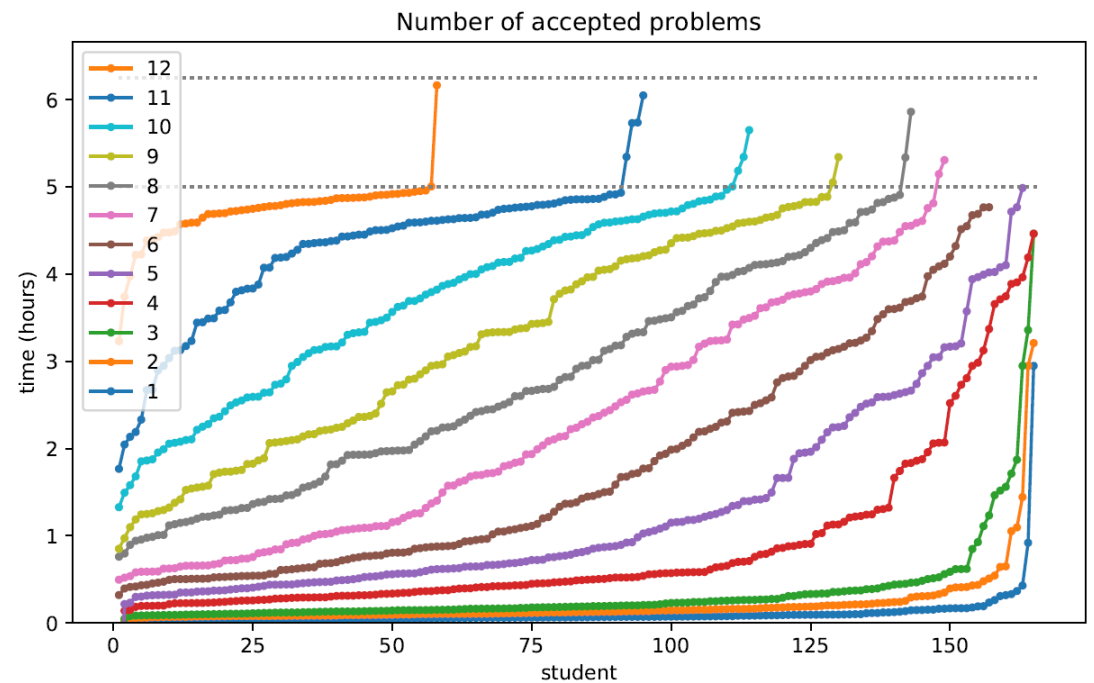
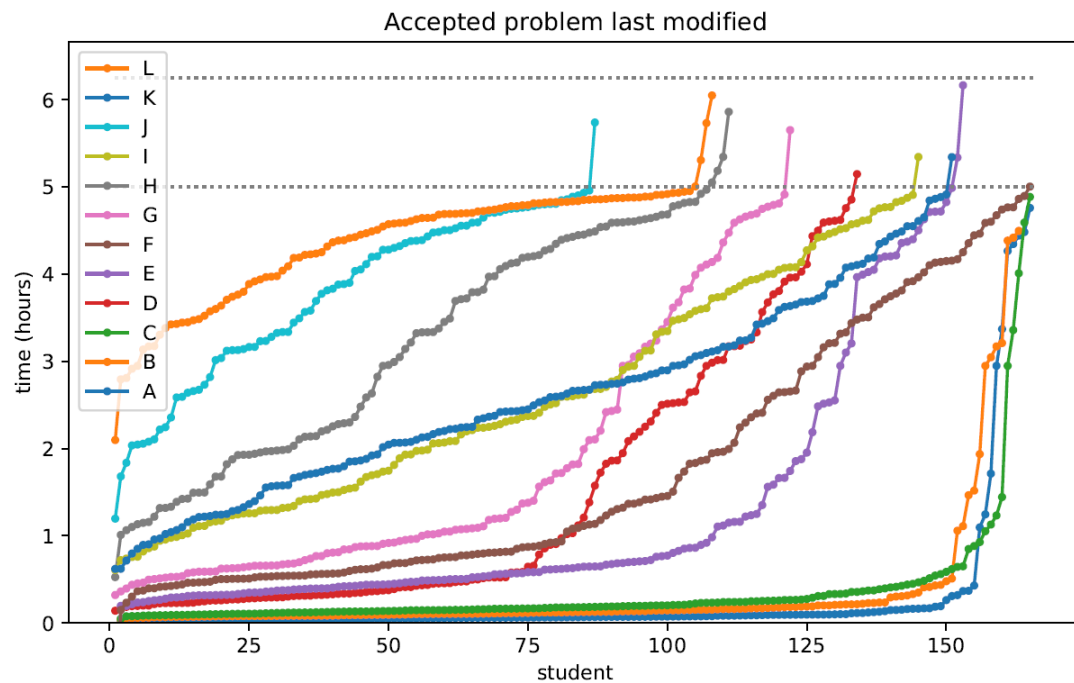
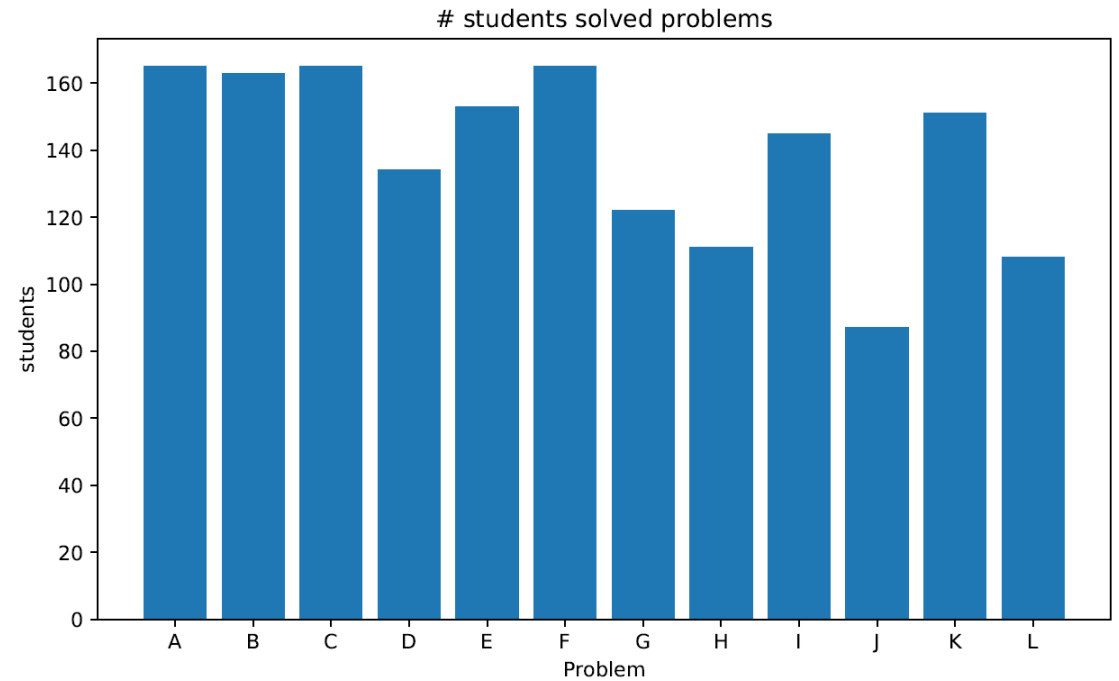
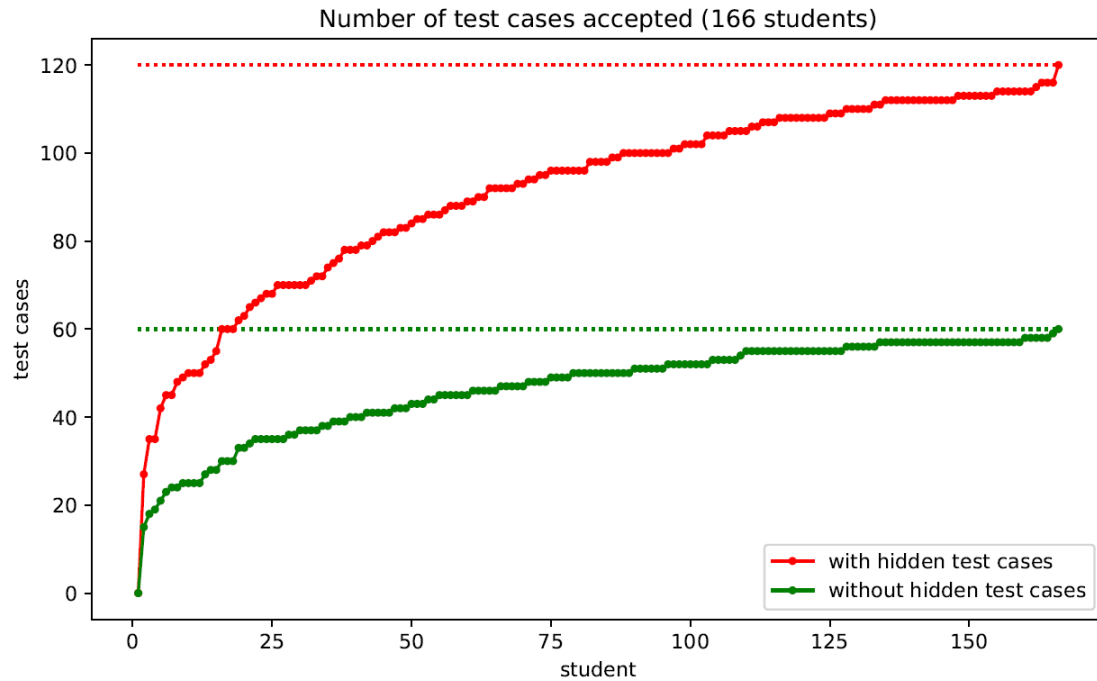
Problem	Point	Navn
A	5	HELLO
B	6	SIGN
C	6	PAIRS
D	6	POINTS
E	7	SET
F	10	REPEATING WORDS
G	10	VALID RANGE
H	10	STARS
I	10	ROTATING VECTOR
J	10	LABELS
K	10	NONNEGATIVE
L	10	SAFE DISTANCE

```
A.py - C:\Users\au121\Desktop\ipsa21\ipsa21exam\A.py (3.10.0)
File Edit Format Run Options Window Help
1 '''
2 HELLO
3
4 Your task is to read a line containing a string S, and to print
5 the string "Hello " followed by S followed by ".".
6
7 Example:
8
9 Input: World
10
11 Output: Hello World.
12 '''
13
14
15 # insert code
16
17 pass
18
19 |
Ln: 19 Col: 0
```

```
run_tests.py - C:\Users\au121\Desktop\ipsa21\ipsa21exam\run_tests.py (3.10.0)
File Edit Format Run Options Window Help
109 def run_test(program, test_in, test_out, verbose, timeout=TIMEOUT):
110     '''Run program with test_in as input. Returns True if and only if
111         the generated output is test_output, and the execution did not
112         generate any errors.
113
114         verbose determines the amount of printed log.
115     '''
116
117     if verbose >= 1:
118         log(f'{program} {test_in} ', end='')
119
120     # Excecute program in subprocess
121
122     with TemporaryFile('w+') as answer_file, \
123         TemporaryFile('w+') as error_file, \
124         open(test_in) as in_file:
125         try:
126             result = subprocess.run(
127                 [PYTHON, program],
128                 input=''.join(line.strip() + '\n' for line in in_file),
129                 #stdin=in_file, # can fail on Mac
130                 stdout=answer_file,
131                 stderr=error_file,
132                 timeout=timeout,
133                 text=True
134             )
135         except Exception as e:
136             exception = str(e)
137         else:
138             exception = ''
139
140
```



# IPSA 2021 Exam





# Udfordringer

- Meget spredte forudsætninger blandt de studerende
- Ny version af sproget hvert år (oktober)
- Forskellige versioner af Python (Python 3.10, 3.9, ..., PyPy, 64/32 bit)
- Forskelligt hardware og OS (Windows, Mac, Linux)
- Forskellige IDE (IDLE, Visual Studio, PyCharm, Spyder, Jupyter, repl.it, emacs...)
- Mange instruktører ikke datalogistuderende (forskellig baggrund)
- Ikke alle Python pakker tilgængelige til semesterstart til nyeste Python
- Logistik omkring studiecafé (Discord godt alternativ)
- +300.000 projekter på PyPI

# Tak til alle for hjælpen med at skabe kurset

- Mathias Rav
- Niels Lauritzen
- Allan Grønlund
- Thomas Vosegaard
- Alle instruktorer, studerende, ...

