

INTRODUKTION

Programmering

Torsdag 4. september 2025

COGNITIVE LOAD THEORY



PROGRAMMERING

← KURSUSKATALOG

Programming

Efterårssemester 2025

ECTS	UNDERVISNINGSFORM >	EKSAMENSFORM >	UNDERVISNINGSSPROG	NIVEAU	STED
5	Fjernundervisning m. seminarer	Mundtlig	Dansk	Master	Aarhus

[link](#)

PROGRAMMERING

Kursusindhold

Kurset giver en introduktion til programmering.

Grundlæggende programmeringsbegreber og -teknikker introduceres ved hjælp af pseudokode samt programmering i Python. Introduktionen vil suppleres med eksempler fra blokprogrammering i Scratch og unplugged programmering.

Programmeringsbegreberne i Python omfatter datatyper, operatorer, variabler, kontrolstrukturer, betingelser, løkker, funktioner, rekursion, scope og exceptions.

Udvikling af programmer tager udgangspunkt i fagdidaktiske metoder samt fokus på test og fejlfinding.

PROGRAMMERING

Kursusindhold

Kurset giver en **introduktion til programmering**.

Grundlæggende programmeringsbegreber og -teknikker introduceres ved hjælp af **pseudokode** samt programmering i **Python**. Introduktionen vil suppleres med eksempler fra **blokprogrammering i Scratch** og **unplugged programmering**.

Programmeringsbegreberne i Python omfatter datatyper, operatorer, variabler, kontrolstrukturer, betingelser, løkker, funktioner, rekursion, scope og exceptions.

Udvikling af programmer tager udgangspunkt i **fagdidaktiske metoder** samt fokus på **test og fejlfinding**.

PROGRAMMERING

Overordnede ideer

Generel introduktion til programmering



PROGRAMMERING

Overordnede ideer

Generel introduktion til programmering

Relevans ift undervisning af andre



PROGRAMMERING

Overordnede ideer

Generel introduktion til programmering

Relevans ift undervisning af andre

Konkrete undervisningsforløb, der kan genbruges efter eventuel tilretning



PROGRAMMERING

Overordnede ideer

Generel introduktion til programmering

Relevans ift undervisning af andre

Konkrete undervisningsforløb, der kan genbruges efter eventuel tilretning

Tilgang til undervisning TOP DOWN og BOTTOM UP



PROGRAMMERING

Overordnede ideer

Generel introduktion til programmering

Relevans ift undervisning af andre

Konkrete undervisningsforløb, der kan genbruges efter eventuel tilretning

Tilgang til undervisning TOP DOWN og BOTTOM UP

Applikationsorienteret tilgang



PROGRAMMERING

Overordnede ideer

Generel introduktion til programmering

Relevans ift undervisning af andre

Konkrete undervisningsforløb, der kan genbruges efter eventuel tilretning

Tilgang til undervisning TOP DOWN og BOTTOM UP

Applikationsorienteret tilgang

Udvikling af pseudokode og den algoritmiske tankegang



PROGRAMMERING

Overordnede ideer

Generel introduktion til programmering

Relevans ift undervisning af andre

Konkrete undervisningsforløb, der kan genbruges efter eventuel tilretning

Tilgang til undervisning TOP DOWN og BOTTOM UP

Applikationsorienteret tilgang

Udvikling af pseudokode og den algoritmiske tankegang

Variation med unplugged programmering, herunder bodygramming



PROGRAMMERING

4. september –fysisk
11. september –online
25. september –online

2. oktober –fysisk
9. oktober –online
23. oktober –online

30. oktober –fysisk
6. november –online
20. november –online

27. november –fysisk
4. december –online

PROGRAMMERING

4. september – fysisk
11. september – online
25. september – online

2. oktober – fysisk
9. oktober – online
23. oktober – online

30. oktober – fysisk
6. november – online
20. november – online

27. november – fysisk
4. december – online

Eksamen

11. og 12. december – fysisk

PROGRAMMERING

Fysiske møder

9:00 – 9:30	Opsamling fra sidst
9:30 – 10:45	Undervisning – oplæg og gruppearbejde
10:45 – 11:00	Pause
11:00 – 12:15	Undervisning – oplæg og gruppearbejde
12:15 – 13:00	Frokost
13:00 – 14:15	Undervisning – oplæg og gruppearbejde
14:15 – 14:30	Pause
14:30 – 15:45	Undervisning – oplæg og gruppearbejde
15:45 – 16:00	Afrunding

Online møder

15:00 – 17:00	Gruppearbejde med vejledning – evt korte oplæg
---------------	--

PROGRAMMERING

Gruppearbejde



PROGRAMMERING

Gruppearbejde

Grupper på 2 –4



PROGRAMMERING

Gruppearbejde

Grupper på 2 –4

Afleveringsopgaver

Efter de første seks online møder

4. september –fysisk

11. september –online - aflevering

25. september –online - aflevering

2. oktober –fysisk

9. oktober –online - aflevering

23. oktober –online - aflevering

30. oktober –fysisk

6. november –online - aflevering

20. november –online - aflevering

27. november –fysisk

4. december –online

PROGRAMMERING

Gruppearbejde

Grupper på 2 –4

Afleveringsopgaver

Efter de første seks online møder

Gruppeafleveringer

4. september –fysisk

11. september –online - aflevering

25. september –online - aflevering

2. oktober –fysisk

9. oktober –online - aflevering

23. oktober –online - aflevering

30. oktober –fysisk

6. november –online - aflevering

20. november –online - aflevering

27. november –fysisk

4. december –online

PROGRAMMERING

Gruppearbejde

Grupper på 2 –4

Afleveringsopgaver

Efter de første seks online møder

Gruppeafleveringer

Forudsætning for eksamen

4. september –fysisk

11. september –online - aflevering

25. september –online - aflevering

2. oktober –fysisk

9. oktober –online - aflevering

23. oktober –online - aflevering

30. oktober –fysisk

6. november –online - aflevering

20. november –online - aflevering

27. november –fysisk

4. december –online

PROGRAMMERING

Gruppearbejde

Grupper på 2 –4

Afleveringsopgaver

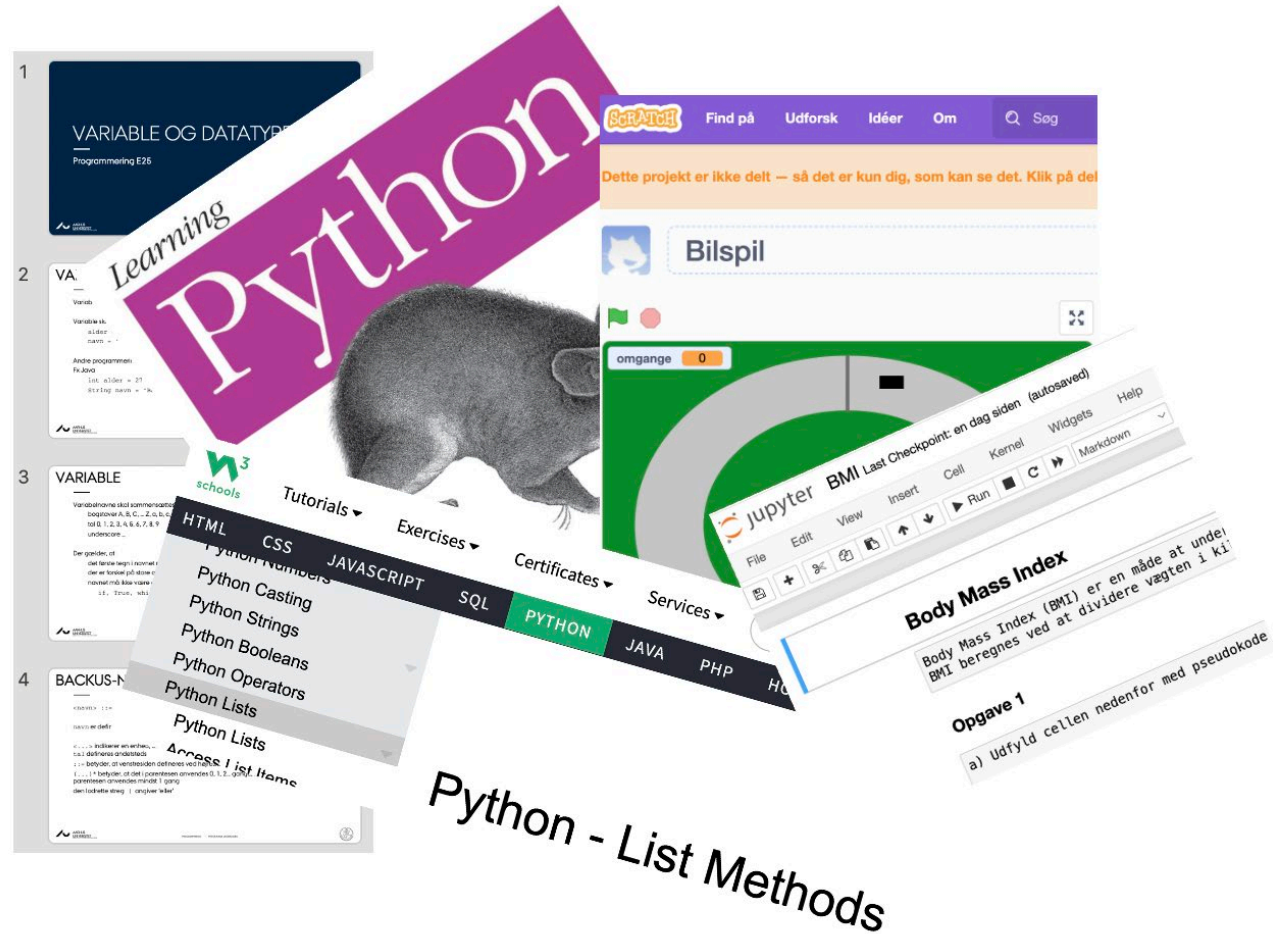
Efter de første seks online møder

Gruppeafleveringer

Forudsætning for eksamen

Materiale

Slides og digitalt materiale



PROGRAMMERING

Gruppearbejde

Grupper på 2 –4

Afleveringsopgaver

Efter de første seks online møder

Gruppeafleveringer

Forudsætning for eksamen

Materiale

Slides og digitalt materiale

Brightspace



Programmering (E25.520254U001.A)



Course Home Content Course Tools ▾ Classlist Zoom Panopto Help



Programmering (E25.520254U001.A)



AARHUS
UNIVERSITET
INSTITUT FOR DATALOGI

PROGRAMMERING | PER SELMANN ANDREASSEN



PROGRAMMERING

Gruppearbejde

Grupper på 2 –4

Afleveringsopgaver

Efter de første seks online møder

Gruppeafleveringer

Forudsætning for eksamen

Materiale

Slides og digitalt materiale

Brightspace

Mail

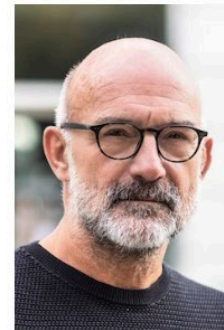
Kontakt

I er altid velkomne til at skrive eller ringe til mig

Per Andreassen

pean@cs.au.dk

2064 7570



PROGRAMMERING

Arbejdsbelastning

5 ECTS svarer til ca 137 arbejdstimer



PROGRAMMERING

Arbejdsbelastning

5 ECTS svarer til ca 137 arbejdstimer

4 fysiske seminarer á 7 timer er 28 timer



PROGRAMMERING

Arbejdsbelastning

5 ECTS svarer til ca 137 arbejdstimer

4 fysiske seminarer á 7 timer er 28 timer

7 online møde á 2 timer er 14 timer



PROGRAMMERING

Arbejdsbelastning

5 ECTS svarer til ca 137 arbejdstimer

4 fysiske seminarer á 7 timer er 28 timer

7 online møde á 2 timer er 14 timer

I alt 42 timer



PROGRAMMERING

Arbejdsbelastning

5 ECTS svarer til ca 137 arbejdstimer

4 fysiske seminarer á 7 timer er 28 timer

7 online møde á 2 timer er 14 timer

I alt 42 timer

Hjemmearbejde (inkl eksamen) $137 - 42 = 95$ timer



PROGRAMMERING

Arbejdsbelastning

5 ECTS svarer til ca 137 arbejdstimer

4 fysiske seminarer á 7 timer er 28 timer

7 online møde á 2 timer er 14 timer

I alt 42 timer

Hjemmearbejde (inkl eksamen) $137 - 42 = 95$ timer

14 uger med ca 10 arbejdstimer pr uge



MOTIVATION

Python

Scratch

Pseudokode

Unplugged programming

Eksemplariske undervisningsforløb

MOTIVATION

Python

Scratch

Pseudokode

let at lære
overskueligt
enkelt
supplerer hinanden

Unplugged programmering

Eksemplariske undervisningsforløb

MOTIVATION

Python

Scratch

Pseudokode

Unplugged programmering

Eksemplariske undervisningsforløb

algoritmisk tankegang
computationel tankegang
springbræt til alle programmeringssprog

forsøg...

MOTIVATION

Python

Scratch

Pseudokode

Unplugged programmering

Eksemplariske undervisningsforløb

variation

på papiret

spillekort

bodygramming - rollespil

MOTIVATION

Python

Scratch

Pseudokode

Unplugged programmering

Eksemplariske undervisningsforløb

lær at programmere for at lære andre det
fagdidaktiske principper
brug meget gerne materialet i egen undervisning

MOTIVATION

Pseudokode

```
antal ord i en tekst
  nulstil antallet
  for hvert tegn i teksten
    hvis tegn er mellemrum
      tæl antallet op med 1
  tæl antallet op med 1
  returner antallet
```

MOTIVATION

Pseudokode

```
antal ord i en tekst
nulstil antallet
for hvert tegn i teksten
    hvis tegn er mellemrum
        tæl antallet op med 1
tæl antallet op med 1
returner antallet
```

Python

```
def antalOrd(tekst):
    antal = 0
    for tegn in tekst:
        if tegn == ' ':
            antal += 1
    antal += 1
    return antal
```

MOTIVATION

Python

```
def antalOrd(tekst):  
    antal = 0  
    for tegn in tekst:  
        if tegn == ' ':  
            antal += 1  
    antal += 1  
    return antal
```

Pascal

MOTIVATION

Python

```
def antalOrd(tekst):  
    antal = 0  
    for tegn in tekst:  
        if tegn == ' ':  
            antal += 1  
    antal += 1  
    return antal
```

Pascal

```
function antalOrd(tekst: string): integer;  
var  
    i, antal: integer;  
begin  
    antal := 0;  
    for i := 1 to length(tekst) do  
        begin  
            if tekst[i] = ' ' then  
                antal := antal + 1;  
        end;  
    antal := antal + 1;  
    antalOrd := antal;  
end;
```


MOTIVATION

Python

```
def antalOrd(tekst):  
    antal = 0  
    for tegn in tekst:  
        if tegn == ' ':  
            antal += 1  
    antal += 1  
    return antal
```

Pascal

```
function antalOrd(tekst: string): integer;  
var  
    i, antal: integer;  
begin  
    antal := 0;  
    for i := 1 to length(tekst) do  
        begin  
            if tekst[i] = ' ' then  
                antal := antal + 1;  
        end;  
    antal := antal + 1;  
    antalOrd := antal;  
end;
```

MOTIVATION

Python

```
def antalOrd(tekst):  
    antal = 0  
    for tegn in tekst:  
        if tegn == ' ':  
            antal += 1  
    antal += 1  
    return antal
```

Pascal

```
function antalOrd(tekst: string): integer;  
var  
    i, antal: integer;  
begin  
    antal := 0;  
    for i := 1 to length(tekst) do  
        begin  
            if tekst[i] = ' ' then  
                antal := antal + 1;  
        end;  
    antal := antal + 1;  
    antalOrd := antal;  
end;
```

MOTIVATION

Python

```
def antalOrd(tekst):  
    antal = 0  
    for tegn in tekst:  
        if tegn == ' ':  
            antal += 1  
    antal += 1  
    return antal
```

indrykninger (= indentations)

Pascal

```
function antalOrd(tekst: string): integer;  
var  
    i, antal: integer;  
begin  
    antal := 0;  
    for i := 1 to length(tekst) do  
        begin  
            if tekst[i] = ' ' then  
                antal := antal + 1;  
        end;  
        antal := antal + 1;  
    antalOrd := antal;  
end;
```

MOTIVATION

Python

```
def antalOrd(tekst):  
    antal = 0  
    for tegn in tekst:  
        if tegn == ' ':  
            antal += 1  
    antal += 1  
    return antal
```

Pascal

```
function antalOrd(tekst: string): integer;  
var  
    i, antal: integer;  
begin  
    antal := 0;  
    for i := 1 to length(tekst) do  
        begin  
            if tekst[i] = ' ' then  
                antal := antal + 1;  
        end;  
    antal := antal + 1;  
    antalOrd := antal;  
end;
```

MOTIVATION

Python

```
def antalOrd(tekst):  
    antal = 0  
    for tegn in tekst:  
        if tegn == ' ':  
            antal += 1  
    antal += 1  
    return antal
```

indrykninger (= indentations)










Pascal

```
function antalOrd(tekst: string): integer;  
var  
    i, antal: integer;  
begin  
    antal := 0;  
    for i := 1 to length(tekst) do  
        begin  
            if tekst[i] = ' ' then  
                antal := antal + 1;  
        end;  
    antal := antal + 1;  
    antalOrd := antal;  
end;
```

VALG AF PYTHON

TIOBE indexmaj 2025



May 2025	May 2024	Change	Programming Language		Ratings	Change
1	1			Python	25.35%	+9.02%
2	3	▲		C++	9.94%	+0.41%
3	2	▼		C	9.71%	-0.27%
4	4			Java	9.31%	+0.62%
5	5			C#	4.22%	-2.27%
6	6			JavaScript	3.68%	+0.66%
7	8	▲		Go	2.70%	+1.10%
8	7	▼		Visual Basic	2.62%	+0.61%
9	11	▲		Delphi/Object Pascal	2.29%	+1.05%

VALG AF PYTHON



Over flere år...

Programming Language	2025	2020	2015	2010	2005	2000	1995	1990	1985
Python	1	3	7	7	9	25	22	-	-
C++	2	4	4	4	3	2	1	3	11
Java	3	1	2	1	2	3	-	-	-
C	4	2	1	2	1	1	2	1	1
C#	5	5	5	6	10	9	-	-	-
JavaScript	6	7	8	9	11	6	-	-	-
Go	7	13	48	184	-	-	-	-	-
Visual Basic	8	15	94	-	-	-	-	-	-
SQL	9	9	-	-	-	-	-	-	-
Fortran	10	31	31	25	16	19	5	2	6
PHP	12	8	6	3	5	26	-	-	-

VALG AF PYTHON

Men der er et problem med Python...

Python har en dårlig ydeevne mht energi og tid,
da det er et fortolket sprog og ikke kompileret.

Total			
Language	Energy	Language	Time
C	1.00	C	1.00
Rust	1.03	Rust	1.04
C++	1.34	C++	1.56
Java	1.98	Java	1.89
Pascal	2.14	Go	2.83
Lisp	2.27	Pascal	3.02
Swift	2.79	C#	3.14
C#	3.14	Lisp	3.40
Go	3.23	Swift	4.20
JavaScript	4.45	JavaScript	6.52
TypeScript	21.50	PHP	27.64
PHP	29.30	TypeScript	46.20
Ruby	69.91	Ruby	59.34
Python	75.88	Python	71.90

[link](#)

VALG AF PYTHON

Men der er et problem med Python...

Python har en dårlig ydeevne mht energi og tid,
da det er et fortolket sprog og ikke kompileret.

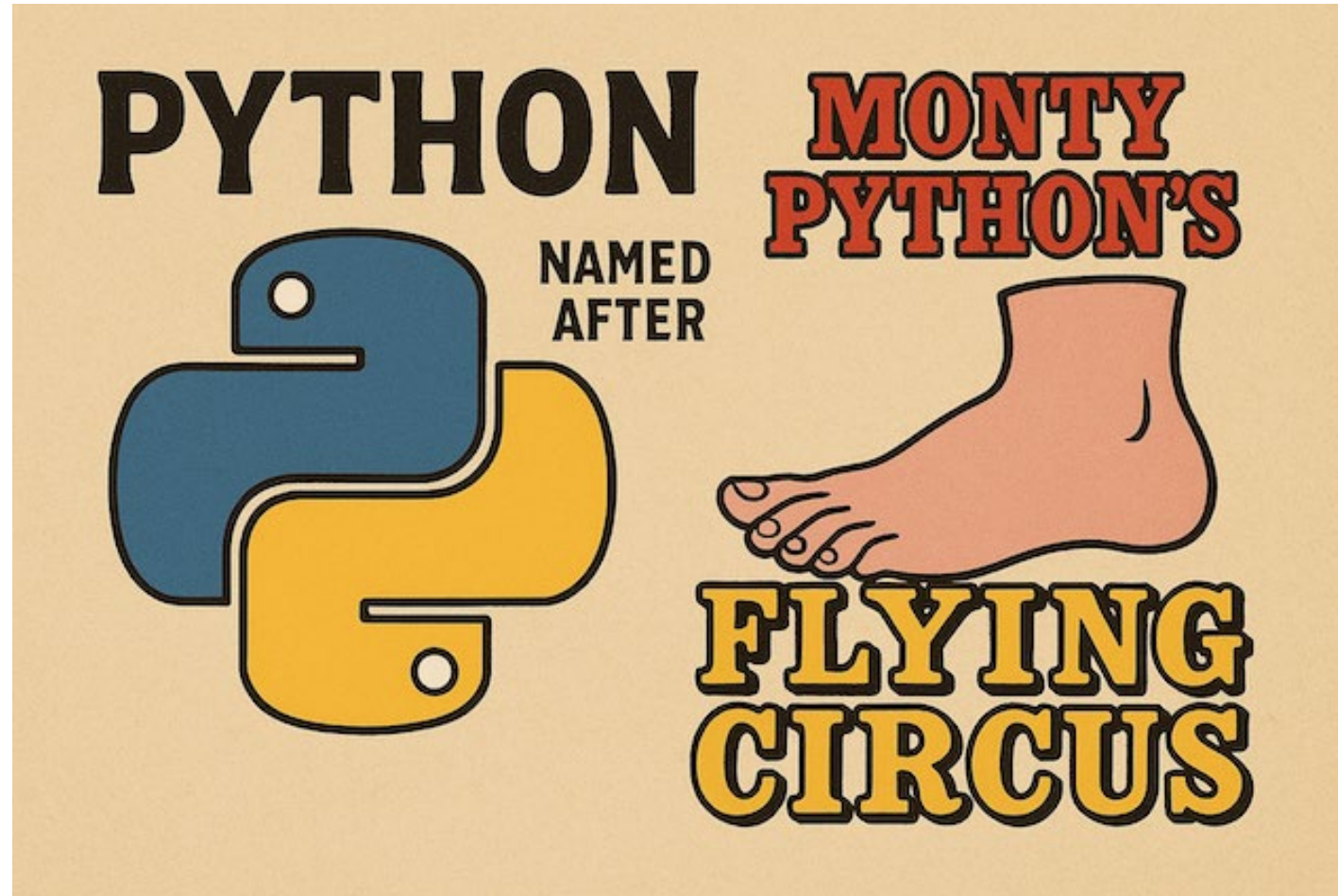
Store fordele med Jupyter Notebook
dokumenter med koder, tekst, billeder, links...



Total			
Language	Energy	Language	Time
C	1.00	C	1.00
Rust	1.03	Rust	1.04
C++	1.34	C++	1.56
Java	1.98	Java	1.89
Pascal	2.14	Go	2.83
Lisp	2.27	Pascal	3.02
Swift	2.79	C#	3.14
C#	3.14	Lisp	3.40
Go	3.23	Swift	4.20
JavaScript	4.45	JavaScript	6.52
TypeScript	21.50	PHP	27.64
PHP	29.30	TypeScript	46.20
Ruby	69.91	Ruby	59.34
Python	75.88	Python	71.90

[link](#)

MONTY PYTHON...



SPØRGESKEMA

Undervisningserfaring

Mellem 6 og 25 år og gennemsnit 14 år

SPØRGESKEMA

Undervisningserfaring

Mellem 6 og 25 år og gennemsnit 14 år

Gengangere

10 af 27 er også tilmeldt *Algoritmer og datastrukturer*

SPØRGESKEMA

Undervisningserfaring

Mellem 6 og 25 år og gennemsnit 14 år

Gengangere

10 af 27 er også tilmeldt *Algoritmer og datastrukturer*

Informatik

14 af 27 underviser i informatik eller andet it-fag

SPØRGESKEMA

Undervisningserfaring

Mellem 6 og 25 år og gennemsnit 14 år

Gengangere

10 af 27 er også tilmeldt *Algoritmer og datastrukturer*

Informatik

14 af 27 underviser i informatik eller andet it-fag

Fag – 18 forskellige

bi, da, di, en, fi, fy, hi, id, iø, ke, ma, me, mu, ps, pæ, re, sa, te

SPØRGESKEMA

Undervisningserfaring

Mellem 6 og 25 år og gennemsnit 14 år

Gengangere

10 af 27 er også tilmeldt *Algoritmer og datastrukturer*

Informatik

14 af 27 underviser i informatik eller andet it-fag

Fag – 18 forskellige

bi, da, di, en, fi, fy, hi, id, iø, ke, ma, me, mu, ps, pæ, re, sa, te

Steder

København, Esbjerg, Odense, Aarhus...

SPØRGESKEMA

Undervisningserfaring

Mellem 6 og 25 år og gennemsnit 14 år

Gengangere

10 af 27 er også tilmeldt *Algoritmer og datastrukturer*

Informatik

14 af 27 underviser i informatik eller andet it-fag

Fag – 18 forskellige

bi, da, di, en, fi, fy, hi, id, iø, ke, ma, me, mu, ps, pæ, re, sa, te

Steder

København, Esbjerg, Odense, Aarhus...

Erfaring med Scratch

Ingen 8

Lidt 16

Meget 2

SPØRGESKEMA

Undervisningserfaring

Mellem 6 og 25 år og gennemsnit 14 år

Gengangere

10 af 27 er også tilmeldt *Algoritmer og datastrukturer*

Informatik

14 af 27 underviser i informatik eller andet it-fag

Fag – 18 forskellige

bi, da, di, en, fi, fy, hi, id, iø, ke, ma, me, mu, ps, pæ, re, sa, te

Steder

København, Esbjerg, Odense, Aarhus...

Erfaring med Scratch

Ingen 8

Lidt 16

Meget 2

Erfaring med Python

Ingen 13

Lidt 13

Meget 0