

# EKSAMENSFORSIDE

## Digital hjemmeeksamen

Emnekode: PY1010	Emnenavn: Python-programmering for beregninger (gjennomføring 2)	
Dato: Fredag 11. september 2020	Tid fra / til: Kl. 09:00 - 12:00	Ant. timer: 3
Ansv. faglærer: Finn Aakre Haugen (97019215, finn.haugen@usn.no)		
Campus: Porsgrunn/digital hjemmeeksamen	Fakultet: Fakultet for teknologi, naturvitenskap og maritime fag	
Antall oppgaver: 20	Antall vedlegg: 0	Ant. sider inkl. forside og vedlegg: 10
Tillatte hjelpemidler: Alle hjelpemidler er tillatte, men du skal ikke kommunisere med andre under eksamen (eksamen er individuell).		
Opplysninger om vedlegg: Det er ingen vedlegg.		
Merknader: Du kan ikke kontakte læreren for å få hjelp til å tolke eller forstå oppgaven. I hver av oppgavene er kun ett av svaralternativene korrekt. Besvarelsen skal være i form av et tekstdokument der du har angitt svaralternativet som en bokstav for hver oppgave, f.eks. «2a» for svaralternativ a på oppgave 2.		

Kryss av for type eksamenspapir

Ruter

Linjer

### Oppgave 1

Hva skriver koden nedenfor i konsollen?

```
1 kursnavn = 'Python for beregninger'  
2  
3 print(kursnavn[:6])
```

Svaralternativer:

- a) 'Python for beregninger Python for beregninger Python for beregninger Python for beregninger Python for beregninger Python for beregninger Python for beregninger'
- b) 'Python for beregninger:6'
- c) 'Python'
- d) ''

### Oppgave 2

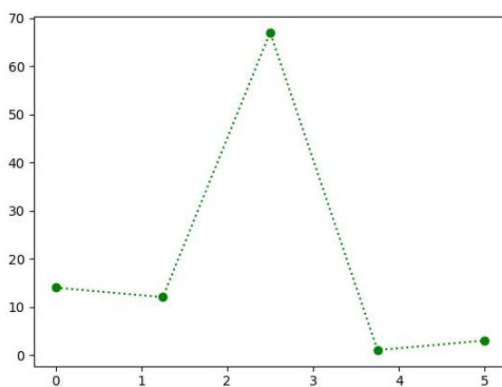
Hvilken datastruktur (list, tuppel og array) egner seg best om man skal ha en samling av verdier av forskjellige datatyper, som man ønsker å kunne redigere på et senere tidspunkt? F.eks. verdiene: 1996, 'Toyota', 1523.15.

Svaralternativer:

- a) Tuppel
- b) List
- c) Array
- d) Alle egner seg like mye.

### Oppgave 3

Hvilken av kodesnuttene nedenfor genererer plottet som er vist i figuren?



Svaralternativer:

- a) plt.plot(x, y, 'g\*')
- b) plt.plot(x, y, 'go')
- c) plt.plot(x, y, 'g-')
- d) plt.plot(x, y, 'go:')

#### Oppgave 4

Hva skriver programmet nedenfor til konsollen?

```
1 def func(a, b = 2):  
2     return a*b  
3 print(func(3), func(3,3))
```

Svaralternativer:

- a) Programmet gir en feilmelding.
- b) 3 (3,3)
- c) 3 6
- d) 6 9

#### Oppgave 5

Koden vist nedenfor gir en feilmelding. Hva er årsaken til feilen?

```
1 import matplotlib.pyplot as plt  
2 import numpy as np  
3  
4 y = [1, 2, 3, 4, 5]  
5 x = np.linspace(1,5,10)  
6  
7 plt.plot(x, y, 'r*--')  
8 plt.show()
```

Svaralternativer:

- a) Feilen ligger i at x ikke starter på null, men 1.
- b) Feilen kommer av at det er ulikt antall verdier i x og y.
- c) Feilen kommer av 'r\*--' parameteren i plot, den fungerer kun på kontinuerlige funksjoner.
- d) Årsaken til feilen er at y parameteren er en liste istedenfor en array.

#### Oppgave 6

Gitt koden nedenfor. Hvilket av alternativene vil returnere [3 4 5]?

```
1 import numpy as np  
2  
3 data = np.array([1,2,3,4,5,6])
```

Svaralternativer:

- a. data[2:4]
- b. data[2:5]
- c. data[3:5]
- d. data[3:6:1]

**Oppgave 7**

Gitt koden nedenfor. Hva skrives til konsollen?

```
1 data = (1,9,3,4,5)
2
3 data[1] = 2
4
5 print(data[1])
```

Svaralternativer:

- a) 2
- b) 9
- c) Koden gir en feilmelding.
- d) 1

**Oppgave 8**

Gitt koden nedenfor. Hva skrives til konsollen?

```
1 x = 1
2
3 def f(y):
4     x = 2*y
5     print(2*x)
6
7 f(x)
8 print(x)
```

Svaralternativer:

- a) 4  
1
- b) 2  
1
- c) 4  
2
- d) 2  
2

**Oppgave 9**

Gitt koden til nedenfor. Hvilken påstand er sann?

```
1 import numpy as np
2
3 n = 10
4
5 data1 = np.linspace(0,1,n)
6 data2 = np.arange(0,1,1/n)
```

Svaralternativer:

- a) data1 og data2 har lik lengde og de inneholder de samme verdiene.
- b) data1 og data2 har lik lengde, men inneholder ikke de samme verdiene.

- c) data1 og data2 har ikke lik lengde, men inneholder samme verdier bortsett fra siste.  
d) data1 og data2 har hverken lik lengde eller inneholder samme verdier.

### Oppgave 10

Gitt koden nedenfor. Hva vil skje?

```
1 def f(x):  
2     return 1/x  
3  
4 a = 0  
5  
6 print(f(a))
```

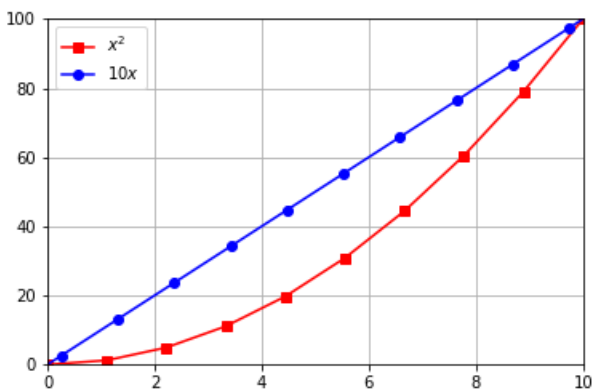
Svaralternativer:

- a) Programmet kjører til slutten, men funksjonen returnerer ingen ting siden vi deler på 0.  
b) Programmet kjører frem til vi forsøker å dele på 0 og gir deretter en feilmelding.  
c) Programmet vil ikke kjøre og gir en feilmelding.  
d) Programmet skriver ut verdien None

### Oppgave 11

Nedenfor ser du starten av koden til et plott og det tilhørende plottet. Avgjør hvilken av kodesnuttene nedenfor som vil generere plottet.

```
1 import numpy as np  
2 import matplotlib.pyplot as plt  
3  
4 x1 = np.linspace(0,10,10)  
5 y1 = x1**2  
6  
7 x2 = np.linspace(-5,15,20)  
8 y2 = 10*x2
```



Svaralternativer:

a)

```
9 plt.plot(x1, y1, 'r-s')
10 plt.plot(x2, y2, 'b-o')
11 plt.xlim(0,10)
12 plt.ylim(0,100)
13 plt.grid()
14 plt.legend(labels=('$x^2$', '$10x$'))
15 plt.show()
```

b)

```
9 plt.plot(x1, y1, 'r-')
10 plt.plot(x2, y2, 'b-')
11 plt.grid()
12 plt.legend(labels=('$x^2$', '$10x$'))
13 plt.show()
```

c)

```
9 plt.plot(x1, y1, 'b-c')
10 plt.plot(x2, y2, 'r-s')
11 plt.grid()
12 plt.legend(labels=('$10x$', '$x^2$'))
13 plt.show()
```

d)

```
9 plt.plot(y1, x1, 'b-o')
10 plt.plot(y2, x2, 'r-s')
11 plt.xlim(-5,15)
12 plt.ylim(0,100)
13 plt.grid()
14 plt.legend(labels=('$x^2$', '$10x$'))
15 plt.show()
```

## Oppgave 12

Hva skriver koden nedenfor til konsollen?

```
1 import numpy as np
2
3 A = [1,2,3]
4 B = np.array([4,5,6])
5
6 print(A + [3.4])
7 print(B + [3.4])
```

Svaralternativer:

a)

[4.4, 5.4, 6.4]

[7.4 8.4 9.4]

b)

[4.4, 5.4, 6.4]

[4. 5. 6. 3.4]

c)

[1, 2, 3, 3, 4]

[4. 5. 6. 3. 4.]

d)

[1, 2, 3, 3.4]

[7.4 8.4 9.4]

### Oppgave 13

Hvilket av følgende funksjonskall setter størrelsen på et plott til A x B tommer (inches)?

*Svaralternativer:*

a) plt.size(A, B)

b) plt.figure(figsize=(A, B))

c) plt.adjust(size=(A,B))

d) plt.dimensions(A,B)

### Oppgave 14

Hva skriver koden nedenfor til konsollen?

```
1  a = 5
2  b = 0
3
4  if( a > b ):
5      b = a
6  else:
7      b = 2
8  if ( a!=b ):
9      b = 3
10 elif( a == b):
11     b = 1
12
13 print(b)
```

*Svaralternativer:*

a) 5

b) 2

c) 3

d) 1

### Oppgave 15

Hvilken logisk (boolsk) operator har nedenstående sannhetstabell?

A	B	
True	True	True
True	False	True
False	True	True
False	False	False

Svaralternativer:

- a) and
- b) or
- c) not
- d) xor

### Oppgave 16

Hva skrives til konsollen når nedenstående program kjøres?

```
1 tall = [31,14,15,92,65]
2
3 for i in range(1, len(tall)):
4     if tall[i-1] < tall[i]:
5         print(tall[i])
```

Svaralternativer:

- a)  
14  
15
- b)  
15  
65
- c)  
15  
92
- d)  
31  
15

### Oppgave 17

Hva skrives til konsollen når nedenstående program kjøres?



```
1 import numpy as np
2
3 data = np.array([2,7,1,8,2,8])
4
5 biggest = 0
6
7 while biggest < 59:
8     data *= 2
9     biggest = np.max(data)
10
11 print(biggest)
```

*Svaralternativer:*

- a) 60
- b) 63
- c) 59
- d) 64

### Oppgave 18

Avgjør hvilken av påstandene nedenfor som ikke er sann.

*Svaralternativer:*

- a) For gjentatte funksjonsberegninger er ofte beregninger med løkker mer effektivt enn vektoriserte beregninger.
- b) En while-løkke kan ha «uendelig» mange iterasjoner dersom dens betingelse aldri endres til False.
- c) Preallokering av arrayer er mer effektivt enn iterativ utviding.
- d) For-løkker bruker ofte range-funksjonen for å forhåndsdefinere antall iterasjoner

### Oppgave 19

Hvilken av kodelinjene nedenfor kan en benytte for å lese en excel fil med navn data.xlsx?

*Svaralternativer:*

- a) `pd.read_excel(data.xlsx, delimiter = '')`.
- b) `pd.read_excel('data')`
- c) `pd.read_excel('data.xlsx')`
- d) `pd.read_excel('data.xlsx', delimiter = '')`

### Oppgave 20

Nedenfor er et bilde av filen «datafil.txt». Hvilket av programmene nedenfor henter ut de to søylene med data og legger dem henholdsvis i variablene «time» og «value»?

```
# Important measurements
# Time, Value
0.000,1.000
0.222,0.952
0.444,0.821
0.667,0.641
0.889,0.454
1.111,0.291
1.333,0.169
1.556,0.089
1.778,0.042
2.000,0.018
```

*Svaralternativer:*

a)

```
1 import numpy as np
2
3 data = np.loadtxt('datafil.txt', delimiter=',')
4
5 time = data[:,0]
6 value = data[:,1]
```

b)

```
1 import numpy as np
2
3 data = np.loadtxt('datafil.txt')
4
5 time = data[0]
6 value = data[1]
```

c)

```
1 import numpy as np
2
3 data = np.loadtxt('datafil.txt', delimiter=' ')
4
5 time = data[0]
6 value = data[1]
```

d)

```
1 import numpy as np
2
3 data = np.loadtxt('datafil.txt', delimiter=',')
4
5 time = data.iloc[0]
6 value = data.iloc[1]
```