ⁱ Forside

UNIVERSITY OF OSLO
Faculty of mathematics and natural sciences

Mid-term exam IN1900, MAT-IN1105, IN-KJM1900
Date: October 13, 2021
Attachments: None
Permitted aids: None

- The exam set has 22 multiple choice questions. The maximum score is 25.
- All code in the question text is written in Python 3.
- Most of the questions have one correct answer. A correctly marked answer gives 1 point, wrong or missing answer gives 0 points. There are no negative points for wrong answers, so one should always mark an answer.
- Questions 15 and 16 have 4 answers each. It is given 0.5 points for each correct answer and 0 points for wrong or unmarked answers, maximum 2 points total for each of the two questions.
- Question 20 has two correct answers. It is given 1 point for each correctly marked answer, -1 for incorrectly marked answers, and 0 for missing answers. The maximum score on the question is 2 and the minimum score is 0. One should always mark at least one answer.

1 Hva skrives ut?

What is printed in the terminal window when the following code is run?

x = 4
y = x
x = 3
print(x, y)
Select one alternative:

○ 3 4

○ 4 4

○ An error message

○ 3 3

Maximum marks: 1

2 Hva skrives ut?

What is printed in the terminal window when the following code is run?
a = "Hello, world!"
print(a, type(a))
Select one alternative:

○ An error message

○ Hello, world! type(Hello, world!)

○ Hello, world! <class 'str'>

○ Hello, world! Hello, world!

Maximum marks: 1

3 Hva skrives ut?

What is printed when the following code is run?
F = 80
C = (F-32)*5.0/9
print(type(F),type(C))

Select one alternative:

○ <class 'float'> <class 'float'>

○ <class 'int'> <class 'int'>

○ <class 'int'> <class 'float'>

○ 80, 26.666

Maximum marks: 1

4 Hva skrives ut?

What is printed when the following code is run?

```
a = []
for i in range(7):
    a.append(i*3)
print(len(a), a[-1])
```

Select one alternative:

- ○ 6 18

- ○ An error message

- ○ 7 21

- ○ 7 18

Maximum marks: 1

5 Ny oppgave

What is printed in the terminal window when this code is run?

```
a = [1,2,3,4]
b = []
for i in range(5):
    b.append(a*i)
print(len(b[-1]))
```

Select one alternative:

- ○ 4

- ○ 25

- ○ An error message

- ○ 16

Maximum marks: 1

6 Hva skrives ut?

What is printed when this code is run?

```
s = 0
for i in range(1,9,2):
    s += 1
print(s)
```

Select one alternative:

- ◯ 9

- ◯ 5

- ◯ 8

- ◯ 4

Maximum marks: 1

7 Hva skrives ut?

What is printed when this code is run?

```
def f(x):
    return 2*x**2 - 4*x
y = 3
x = 2
print(f(2*y))
```

Select one alternative:

- ◯ 24

- ◯ 0

- ◯ An error message

- ◯ 48

Maximum marks: 1

8 Hva skrives ut?

What is printed in the terminal window when this code is run?

```
def g(x):
    return x**3 + x**2
def f(x,y):
    return x + 2*y
x = 1
print(f(x,g(x)))
```

Select one alternative:

○ An error message

○ 5

○ 4

○ 3

Maximum marks: 1

9 Hva skrives ut?

What is printed in the terminal window when this code is run?

```
def squared_sum(n):
    s = 0
    for i in range(n):
        s += i**2
    return s

def test_squared_sum():
    expected = 1 + 4 + 9 + 16
    computed = squared_sum(4)
    msg = f'expected {expected}, got {computed}'
    assert expected == computed, msg

test_squared_sum()
```

Select one alternative:

○ Success: expected 30, got 30

○ AssertionError: expected 30, got 14

○ Nothing is printed

○ NameError: name 'n' is not defined

Maximum marks: 1

10 Hva skrives ut?

What is printed in the terminal window when this code is run?

numbers = [3, 5, 2, 4, 1]

```
n = len(numbers)
for i in range(n):
  for j in range(n-1):
    if numbers[j] > numbers[j+1]:
      tmp = numbers[j]
      numbers[j] = numbers[j+1]
      numbers[j+1] = tmp

print(numbers)
```

Select one alternative:

○ [5, 4, 3, 2, 1]

○ [3, 5, 2, 4, 1]

○ IndexError: list index out of range

○ [1, 2, 3, 4, 5]

○ 5

Maximum marks: 1

11 Hva skrives ut?

What is printed in the terminal window when this code is run?

```
x = 2
y = 1
a = False
b = (x > 1 and y <= 2)
print(a or b)
```

Select one alternative:

○ True

○ An error message

○ False or True

○ False

Maximum marks: 1

12 Hva skrives ut?

What is printed in the terminal window when this code is run?
```
import numpy as np
w = np.linspace(0,3,31)
print(w[-1:0:-4])
```

Select one alternative:

○ IndexError: index -1 is out of bounds for axis 0 with size 31

○ [3. 2.6 2.2 1.8 1.4 1. 0.6 0.2]

○ [0.2 0.6 1. 1.4 1.8 2.2 2.6 3.]

○ []

Maximum marks: 1

13 Hva skrives ut?

What is printed in the terminal window when this code is run?

```
for i in range(2,7,2):
   print(i,end=' ')
   for j in range(1,i):
      print(j,end=' ')
```

The argument "**end = ' '** " to the print-function makes each print end with a space (' ') instead of of a linebreak.

Select one alternative:

○ 2 1 2 4 1 2 3 4 6 1 2 3 4 5 6

○ 2 1 4 1 2 3 6 1 2 3 4 5

○ An error message

○ 2 4 6 1 2 3 4 5 6

Maximum marks: 1

14 Hvilken linje manger?

Which line needs to be insterted for the test to pass?

```
def odd_sum(n):
    # Insert missing line here
    return sum(s)


def test_odd_sum():
    expected = 1 + 3 + 5
    computed = odd_sum(5)
    msg = f'expected {expected}, got {computed}'
    assert expected == computed, msg


test_odd_sum()
```

The function **sum**, which is used inside **odd_sum,** is a built-in function in Python. It takes a list, array, or similar object as argument, and returns the sum of its elements.

Select one alternative:

○ s = list(range(0,n+1,2))

○ s = list(range(n+1,2))

○ s = list(range(1,n+1))

○ s = list(range(1,n+1,2))

Maximum marks: 1

15 Boolske uttrykk

What is the value of the following boolean expressions?
Please match the values:

	True	Undefined/gives error	False
True and (False or True)	○	○	○
not (False or True)	○	○	○
False or (False and True)	○	○	○
not (True == True)	○	○	○

0.5 points for each correct answer. Maximum 2 points on the question.

Maximum marks: 2

16 Lister og boolske uttrykk

What is the value of the boolean expressions when the following code has been run?

```
def multiply(a,b):
    return [a*b_i for b_i in b]


x = [0,1,2,3,4]
y = multiply(2,x)
z = 2*x
```

Please match the values:

	False	True	Undefined/gives error
y[0] == z[0]*2	○	○	○
len(y) == len(z)	○	○	○
y[-1] > z[-1]	○	○	○
y == z	○	○	○

0.5 points for each correct answer. Maximum 2 points on the question.

Maximum marks: 2

17 I hvilken linje feiler koden?

In what line will this code stop and print an error message?

```
import numpy as np
x = np.linspace(0,2,21)
y = np.zeros(len(x))

def f(x):
    y = x**2 - 4
    return y

for i in range(len(x)+1):
    y[i] = f(x[i])
```

Select one alternative:

○ y[i] = f(x[i])

○ y = x**2 - 4

○ y = np.zeros(len(x))

○ for i in range(len(x)+1):

Maximum marks: 1

18 Hvilket funksjonskall?

The function **midpoint(f,a,b,n)** applies the midpoint rule to estimate the integral of a function f on the interval a<x<b.

```
def midpoint(f, a, b, n=100):
    h = float(b-a)/n
    result = 0
    for i in range(n):
        result += f((a + h/2.0) + i*h)
    result *= h
    return result
```

We want to use the function to estimate the integral
$$A = \int_0^1 x^3 - 4x - 2$$

Which function call is correct?
Select one alternative:

- ○ A = midpoint(lambda x: x**3-4*x-2,0,1)

- ○ A = midpoint(f(x)=x**3-4*x-2,0,1)

- ○ A = midpoint(x**3-4*x-2,0,1)

- ○ A = midpoint(f=x**3-4*x-2,0,1)

Maximum marks: 1

19 **Hvilket funksjonskall?**

The function **diff(f,x,h)** estimates the derivative of a function f in the point x:

```
def diff(f,x,h=1e-4):
    return (f(x+h)-f(x))/h
```

We want to use the function to estimate the derivative of the function
$$f(x) = sin(2\pi x)e^{-x}$$
in the point x=4.

Which of the following codes is correct? You can assume that the line
from math import *
has been run before all the alternatives.

Alternative A:
```
def f(x):
    return sin(2*pi*x)*exp(-x)
df = diff(f(x),4)
```

Alternative B:
```
def f(x):
    return sin(2*pi*x)*exp(-x)
df = diff(f,4)
```

Alternative C:
```
def f(x):
    return sin(2*pi*x)*exp(-x)
x = 4
df = diff(f(x))
```

Alternative D:
```
x = 4
df = f(sin(2*pi*x*exp(-x),x)
```

Select one alternative:

○ Alternative C

○ Alternative B

○ Alternative A

○ Alternative D

Maximum marks: 1

20 Beregning av en sum

We want to write a Python function which computes the sum

$$L(x, n) = \sum_{i=1}^{n} \frac{x^i}{i}$$

Two of these codes are correct. Which ones?

Alternative A:
```
def L(x,n):
    s=0
    for i in range(1,n+1):
        s += x**i/i
    return s
```

Alternative B:
```
def L(x,n):
    s=0
    for i in range(1,n):
        s += x**i/i
    return s
```

Alternative C:
```
def L(x,n):
    return sum([x**(i+1)/(i+1) for i in range(n)])
```

Alternative D:
```
def L(x,n):
    return sum([x**i/i for i in range(n)])
```

The function **sum**, which is used in alternatives C and D, is a built-in function in Python. It takes a list, array, or similar object as argument, and returns the sum of its elements.

Select two alternatives:

- ☐ Alternative D is correct.

- ☐ Alternative A is correct.

- ☐ Alternative B is correct.

- ☐ Alternative C is correct.

One point for each correct answer, minus one point for a wrong answer. Maximum two points total on the question. The minimum total score for the question is zero, so one should always mark at least one alternative.

Maximum marks: 2

21 Hvilken kode er riktig?

The file 'messy_records.txt' contains years, times, and names for world records on 100 meter dash:

2007 9.74 Asafa Powell
1996 9.84 Donovan Bailey
2008 9.69 Usain Bolt
1991 9.86 Carl Lewis
1994 9.85 Leroy Burrell
1999 9.79 Maurice Greene
2009 9.58 Usain Bolt
2002 9.78 Tim Montgomery
2005 9.77 Asafa Powell

We want to write a Python program which reads such a file and prints the line with the lowest time (the current world record). For the lines above the program shall print the line

2009 9.58 Usain Bolt

The code below is meant to solve this task, but the contents of the for-loop is missing.

```
infile = open('messy_records.txt')
fastest = 100.0
for line in infile:
    #Missing lines
    #go here

print(record)
```

Which of the following alternatives needs to be added inside the for-loop for the code to work. You can assume that all the alternatives have the right indentation to fit inside the for-loop.

Alternative A:
```
words = line.split()
if words[0] < fastest:
    fastest = t
    record = line
```

Alternative B:
```
words = line.split()
t = float(words)
if t == fastest:
    fastest = t
    record = line
```

Alternative C:
```
t = float(line.split()[1])
if t < fastest:
    fastest = t
    record = line
```

Alternative D:

```
    t = float(line.split()[1])
    if t < fastest:
        record = t
```

Select one alternative:

○ Alternative A is correct.

○ Alternative B is correct.

○ Alternative C is correct.

○ Alternative D is correct.

Maximum marks: 1

22 Hvilken påstand er riktig?

The two functions **sum_while(x,n)** and **sum_for(x,n)** both compute a sum sum, using a while-loop and a for-loop, respectively:

```
def sum_while(x,n):
    s = 0
    i = 1
    while i <= n+1:
        s += x**i/i
        i += 1
    return s


def sum_for(x,n):
    s = 0
    for i in range(1,n+1):
        s += x**i/i
    return s
```

We call the functions as follows:
```
s1 = sum_while(5.0,7)
s2 = sum_for(5.0,7)
```

Which statement is correct?

Select one alternative:

○ The while-loop has an error and will give an error message.

○ The while-loop includes more terms than the for-loop.

○ The two functions return the same value.

○ The last term included in both sums is for i=7.

○ The for-loop has an error and will give an error message.

Maximum marks: 1