Informasjon

Question	Marks	Question type
i		Information or resources
i		Information or resources
i		Information or resources

Automatisk rettet

Question	Marks	Question type
1	4	Matching
2	4	Text Entry
3	6	Text Entry
4	3	Text Entry
5	6	Inline Choice
6	4	Text Entry
7	3	Multiple Choice

Forklaringsspørsmål

Question	Marks	Question type
8	5	Text area
9	5	Text area
10	5	Text area

Kodingsspørsmål

Question	Marks	Question type
11	5	Code compile
12	5	Code compile
13	10	Essay
14	15	Essay

Poeng fra laber

Question	Marks	Question type
15	20	Oral/Sketch

Self-declaration

I hereby declare that the submission I am sending in is my own work.

I have not

- collaborated with other students
- used others' work without acknowledging it
- used my own previous work (assignments/exam submissions) without acknowledging it

If I have used literature, a reference list should include all sources I have used in the assignment, and references should point to this list.

I am aware that violations of these rules are considered cheating and may lead to the cancellation of the exam and/or exclusion.

If you are unsure whether you can stand by this declaration, see the guidelines for source use in written assignments at the University of Bergen, and contact your study advisor/course coordinator.

All exam submissions at the University of Bergen are subject to manual and electronic plagiarism checks.

Note: By continuing, I confirm that I have read the declaration and that the submission I am turning in for this exam is my own work (and only my own work), in full accordance with the declaration above.

Information about the exam

The exam consists of three parts, with a total of 80 points:

Type	Automatically graded questions	Explanatory questions	Coding questions
Points	30	15	35

- If you have problems, you can skip a question and return to it later. To avoid forgetting questions you have skipped, we recommend using the flagging option in the upper right corner.
- You should have started parts 2 and 3 when you have 2.5 hours remaining.
- In part 3, you will have the opportunity to run some of the code you write. However, the entire part 3 will be manually assessed. Therefore, you should not get stuck on a task if your code does not pass the test cases. The examiner is aware of this and will not deduct points for minor mistakes, such as typos in function names from the standard library. However, you must write the code as clearly and correctly as possible to demonstrate that you understand the nuances of the code you are writing.
- Permitted aids for the exam: all written and printed materials.

General advice

- Read the question before answering.
- Work quickly through all the questions in the first round, and return to challenging questions at the end if you have time.
- Help the examiner help you! If you are unsure how to interpret a question, you can write a short comment explaining how you interpret the unclear parts of the question. If you do not remember exactly how to do something in code, you can write a comment explaining what you are trying to do.

Materials

On this page you will find all course notes collected in a single PDF file. They are available to you if you wish to use them. Using them is entirely optional. Standard citation rules apply: if you copy anything from the course notes, it is important that you cite them as a source in your submission.

You can also find the course notes at the bottom left corner throughout the exam. They will open in a new tab.

PS: Have the pages suddenly rotated sideways? Try pressing «R» to rotate them back the right way.

1 x = {'count':3}
y = [8, x]
w = ['strawberry']

Choose the correct data type for the expression

	bool	list	int	set	Error	float	dict	str
{99, 40, 1}	○	○	○	○	○	○	○	○
y[1]	○	○	○	○	○	○	○	○
x.pop(3)	○	○	○	○	○	○	○	○
x['count']/2	○	○	○	○	○	○	○	○
[42, True, x]	○	○	○	○	○	○	○	○
'brush' in 'toothbrush'	○	○	○	○	○	○	○	○
len(w[0])	○	○	○	○	○	○	○	○
f'{123}'	○	○	○	○	○	○	○	○

Maximum marks: 4

2

```
x = 10
y = '10'
u = [True, 2, 3]
w = {
      'cba': 2,
      10: u,
}
s = 'cba'
```

Assume that the code snippet above has been executed, and that one of the statements below is the next statement to run. What is the printed output? If the program crashes, write only Error.

(Remember that apostrophes and quotation marks surrounding strings in the source code are not included in the printed output.)

print(x != y)	
print(not u[0])	
print(y + y)	
print(y[0])	
print(s[-1])	
print(u[1:])	
print(s*w[s])	
print(w[x][2])	

Maximum marks: 4

3

```
def f(y):
    y *= 2
    return y

def g(x):
    x = f(x)
    return f(x) + x
```

Assume that the functions above are defined. What do these expressions evaluate to? (If the expression crashes, write only 'Error')

f(3)	
g(4)	
f(g(1))	

Maximum marks: 6

4

```
def update(x, y):
    if x - y > 50:
        return 2*y
    elif x - y < 50:
        y += 50
    else:
        x -= 50
        print(y)
    return y // x

x = 100
y = x + 50
```

Assume that the code snippet above has been executed, and that the statement below is the next one to run. What is the printed output? If the program crashes, write only Error.

print(update(x, y)) [ ]

Maximum marks: 3

5

```
def f(x, y):
    result = []
    for num in x:
        if num % y == 0:
            result.append(num // y)
        else:
            result.append(num)
    return result

a = [8, 10, 36, 25]
b = f(a, 5)
```

Assume that the code snippet above has been executed, and that one of the statements below is the next statement to run. What is the printed output? If the program crashes, choose only Error.

print(a) Select alternative ([8, 10, 36, 5], [8, 10, 36, 25], Error, [8, 2, 36, 5], [8, 2, 36, 25])

print(b) Select alternative ([8, 2, 36, 25], [8, 10, 36, 5], [8, 2, 36, 5], [8, 2.0, 36, 5.0], Error)

print(f(f(a, 5), 5)) Select alternative (Error, [8, 1, 36, 2], [8, 2.0, 36, 1.0], [8, 2, 36, 1], [8, 1.0, 36, 2.0])

Maximum marks: 6

6

```
x = 20
y = 1
while x > 0:
    x -= 2
    if x % 4 == 0:
        continue
    if x < 10:
        break
     y = 1 - y
```

Assume that the code snippet above has been executed, and that the statement below is the next one to be executed. What is the printed output? If the program crashes, write only Error.

print(x+y) [ ]

Maximum marks: 4

7 This task is about precedence and associativity.
How to place parentheses to get an expression identical to
x or not y and z or w

Select one alternative:

- x or (((not y) and z) or w)
- (x or ((not y) and z)) or w
- x or ((not (y and z)) or w)
- (x or (not (y and z))) or w

Maximum marks: 3

8 A revolutionary committee has seized power. They commissioned the following program to manage the leadership:

```
def foo(lst):
    last = lst[-1]
    for i in range(len(lst) - 1, 0, -1):
        lst[i] = lst[i - 1]
    lst[0] = last

names = ["Bashar", "Robert", "Fidel", "Vladimir", "Kim"]
foo(names)
print(names)
```

- What does the function foo() do? What does the program print out?
- Give the function foo() a better name so that people can easily understand what it does without knowing how it is implemented.

Fill in your answer here

Maximum marks: 5

9 An embryo size is measured in the first weeks of pregnancy using Crown-Rump Length (CRL) - the distance from the top of the head to the end of the spine. A doctor wants to estimate the CRL values for the next trimester based on an expected growth factor. The following function is used to calculate this:

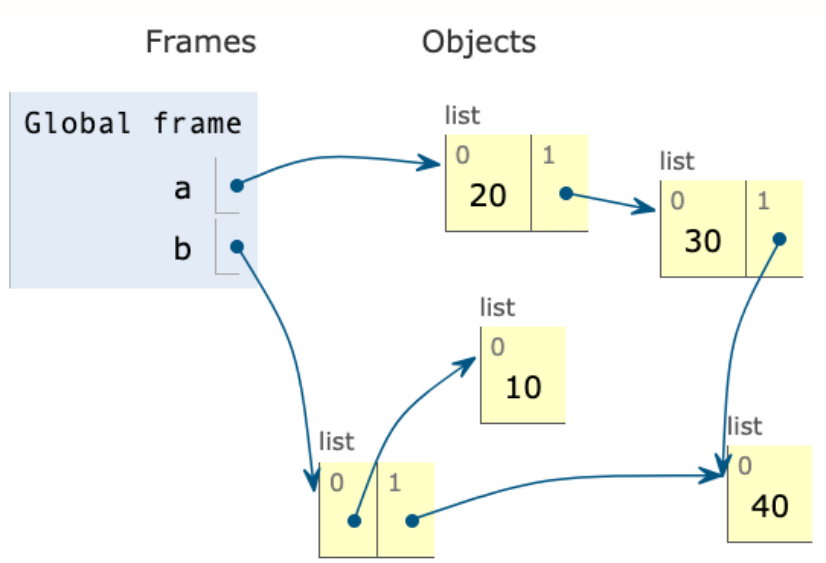
```
def times_factor(crl_measurements, factor):
    new_lst = []
    for sub_lst in crl_measurements:
        new_sub_lst = []
        for num in sub_lst:
            new_sub_lst.append(num*factor)
        new_lst.append(new_sub_lst)
    return new_lst

data = [[12, 16, 21], [38, 45, 52]]
times_factor(data, 10)
print(data)
```

- The output of the run is [[12, 16, 21], [38, 45, 52]], which was not the expected output. What do you believe the expected output should be?
- Why does the program fail to do so?
- How will you fix it?

Fill in your answer here

Maximum marks: 5

10



Write a code snippet so that the memory ends up in the state shown above. For full points, you must solve the task without using helper variables/other variables than **a** and **b**. (The image is taken from Python Tutor.)

Fill in your answer here

Maximum marks: 5

11 Write a function **count_letter(letter, wordlist)** that counts how many times a given letter appears across a list of words. For example, the letter `'b'` appears 5 times in total in the list `['stubborn', 'house', 'cabbage', 'job', 'exam']`.

Skriv svaret ditt her

Test case #	Input	Expected output
1	count_letter('b', ['stubborn', 'house', 'cabbage', 'job', 'exam'])	5
2	count_letter('k', ['bookkeeping', 'lake', 'kickoff', 'tree', 'rucksack'])	7

```
def count_letter(letter, wordlist):
    # Skriv koden her
    # ...

# IKKE rediger under denne linjen!
print(eval(input()))
```

Test code

Maximum marks: 5

12 Continue with the previous task (Task 11).

Write a function **count_letters(data)**.

- **data** is a list of tuples in the form (letter, wordlist)

The function will return a list of integers, where each number indicates how many times the corresponding letter appears in its word list.

Skriv svaret ditt her. Endringer blir lagret automatisk.

Test case #	Input	Expected output
1	count_letters([('b', ['stubborn', 'house', 'cabbage', 'job', 'exam']),('k', ['bookkeeping', 'lake', 'kickoff', 'tree', 'rucksack'])])	[5, 7]

```
def count_letters(data):
    # Skriv koden her
    # ...

# IKKE rediger under denne linjen!
print(eval(input()))
```

Test code

Maximum marks: 5

13

Write a Python program that reads a CSV file containing personal data, and calculates each individual's recommended daily water intake (in milliliters).

Weight-Based (Metric): Multiply the weight (kg) by 30 (ml/kg).
(For example: 70 kg * 30 ml/kg = 2100 ml daily)

Assume that you have a file *health.csv with the content below:*

```
Name;Weight(kg)
Henrik;70
Ingrid;52
Mathias;85
Svanhild;60
```

The printed output of your program should be a list of dictionaries:
[{"Name":"Henrik", "Weight(kg)":70,"Water":2100},
{"Name":"Ingrid", "Weight(kg)":52,"Water":1560},
{"Name":"Mathias", "Weight(kg)":85,"Water":2550},
{"Name":"Svanhild", "Weight(kg)":60,"Water":1800}]

Fill in your answer here

Format | | | | | | Σ |

Words: 0

Maximum marks: 10

14 A logistics company has given you data for their delivery routes and needs you to write a report.

Assume that you have a file *routes.json* with the contents shown on the left side.

In *routes.json*, each data entry contains three pieces of information: the planned route that the driver got from the logistics company, the route that the driver ended up actually taking, and the distance matrix. Each customer is identified by a unique id.

For example, in data entry "2", the planned route specifies that the driver should deliver packages in the order: customer 0, then customer 1, and finally customer 2. However, the driver actually followed a different route: customer 0, then customer 2, and finally customer 1. For simplicity, we display only the data entry "2" and represent the content of the other data entries with ellipses (…).

The indices of the distance matrix represent the customers id. Each row represents the distances from one customer to all others in the route. For instance, in row 0 of the distance matrix for data entry "2", the distance from customer 0 to itself is 0, the distance from customer 0 to customer 1 is 30870, and the distance from customer 0 to customer 2 is 31775.

Assume that *routes.json* contains 100 data entries, and the lists of planned_idx and actual_idx are of equal length in each data entry. For each data entry, calculate the total distance of the planned route and the actual route using the provided distance matrix. Then, compare the two totals and keep track of how many times:

- the actual route is shorter than the planned route,
- the planned route is shorter than the actual route, and
- both routes have equal total distance.

Additionally, the logistics company wants to know the total distance saved by drivers in cases where the actual route is shorter than the planned route. Sum these savings across all such entries.

Fill in your answer here.

Words: 0

Maximum marks: 15

```
{
"1" : { "planned_idx": [...],  "actual_idx": [...], "distances": [...] },
"2" : {
       "planned_idx": [
            0,
            1,
            2
        ],
        "actual_idx": [
            0,
            2,
            1
        ],
        "distances": [
            [
                0,
                30870,
                31775
            ],
            [
                31004,
                0,
                1499
            ],
            [
                32054,
                1625,
                0
            ]
        ]
    },
...,
"100" : { "planned_idx": [...],  "actual_idx": [...], "distances": [...] }
}
```

15 Here, your points from the labs are entered.

Maximum marks: 20