Computational Management Science 1
Fall 2022 Final

registration number: ....
(Do not write your name on the test - just the 7 digit student id number.)

All examples are evaluated using the Python programming language, version 3.10.

1. (6 points) Writing Code

    (a) (3 points, ≤5 minutes) Functions
    Write a small Python function that takes exactly two parameters and returns their sum
    [eg.: `sum(3, 8)` ⇒ 11]. Add a proper docstring with a doctest to receive full points.

    (b) (3 points, ≤5 minutes) Functional programming
    You're given a variable named `sequence` that contains an iterable Python sequence.
    Write a generator expression that yields the square of every odd number in the given
    sequence. Assign the generator expression to a variable.

2. (6 points, ≤10 minutes) Relational Database Design

Design a relational database to store the following information:

We want to store a list of documents for which we know the "name", the "path" at which it is stored and the number of "pages" (integer). In addition, we want to store a list of authors for which we know the "name" and "country".

(a) Every document has a unique author.

In addition, please also give an example on how to store the case of 2 authors and 3 documents, the first two documents being assigned to the first author, the third assigend to the second author.

(b) Every document has none, one or multiple authors. In addition also store the "contribution" (decimal) that gives the proportion the author contributed to the document.

In addition, please also give an example with two authors and two documents. The first document should be written by the 2nd author and the 2nd document should be written jointly by both authors. Assign arbitrary "contribution" to the authorships, but ensure that the sum of contributions for a document is 1.

3. (9 points, ≤10 minutes)

    (a) (2 points)
        Name four properties of Python that make it so successful.

    (b) (2 points)
        Name at least two concepts of object oriented programming.

    (c) (3 points)

        • What is the meaning of the big O?

        • What is the asymptotic complexity of brute forcing the traveling salesman problem?

        • Simplify this term: $O(\frac{3n+5n^2}{e^n} + n!)$

    (d) Finally, name one thing you like about this course and one thing that should be improved in the future (be honest!) (2p).

4. (12 points, ≤10 minutes) Reading and Understanding Code
   What is the output of the following code snippets? Write exactly what the output of each
   snippet is if the snippet is the sole content of a Python file. If the output is an error message,
   it is enough to write "ERROR". If there is no output, write "-".

```python
def sum(a, b, c):
    return a + b + c
sum(4, 2, 3)
```

```python
def circumference(length, width):
    return 2 * length + 2 * width
print(circumference(5, 2))
```

```python
from collections import namedtuple
Point = namedtuple('Point', ('x', 'y'))
A = Point(x=2, y=3)
print(A.x + A.y)
```

```python
s = 'Hello World!'
s.lower()
print(s)
```

```python
l = [4, 2, 3, 1]
print(l.append(0))
```

```python
t = (1, 2, 3, 4)
t.sort()
print(t)
```

5. (9 points, ≤10 minutes)

(a) (3 points)
Below is a function computing the factorial of n:

```python
def factorial(n):
    if n == 1:
        return 1
    else:
        return (n * factorial(n-1))
```

What programming concept is used in this function instead of a loop? Rewrite this function using a loop instead.

(b) (3 points)
What is composition? Describe (as text) an example of a composition. Give a short code example with two classes.

(c) (3 points)
What is an SQL injection? When is such an attack possible in your code and how do you circumvent it?

6. (6 points + 2 bonus, ≤5 minutes) Incomes per region

   You are given a dictionary "region" that has the social security number of inhabitants of a country as a key and the region of the country in which they live as a value. You are given a second dictionary "income" that has the social security number of inhabitants of the same country as the key and their income (float) as a value. You may assume that both dictionaries contain exactly the same set of keys. Write a function `compute_total_income(region, income)` that returns a new dictionary that contains the regions as keys for each such key the sum of income of all inhabitants of that region as value.

   Bonus (2 points): Write a second function `compute_total_income(region, income)` that returns a dictionary containing the average income per region.