

Exercise Sheet 1

Software Architecture for Distributed Embedded Systems, WS 2020/21

Prof. Steinhorst, Dr.-Ing. Mohammad Hamad, M. Sc. Emanuel Regnath

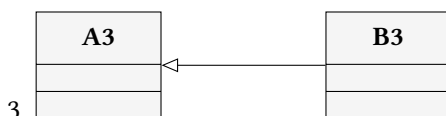
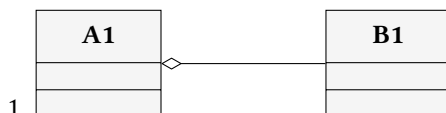
? Exercise 1.1: Unified Modeling Language (UML)

Together with some friends you have founded “AwesomeLights”, a Startup that wants to provide a decentralized indoor lighting solution using smart IoT devices. In contrast to conventional indoor lighting, your system offers to plug-in new switches and lights on the fly and allows dynamic reconfiguration using a smartphone app. Your first task in this new company is to create an UML model of the whole system. In order to learn UML, you decide to solve the following problem questions.

1. You realize that a UML class diagram is about drawing boxes and connecting them with lines. Which information is specified in the 3 sections of a class box? Give an example for each section when modeling a light switch.

1.
2.
3.

2. Which four types of object relations are shown in the following? Name each relation and give an example for the classes A and B that would have this relation. Use the class Human as A or B in each example. What is the difference between the first two relations?



3. How are private, public and protected attributes represented in a class diagram?
4. Which notations specify valid ranges of multiplicity?

☐ 0..

☐ -3

☐ *4

☐ 3-5*

☐ 1..2

☐ 3,*

☐ 3...*

☐ 3..3

☐ 3,4,6

? Exercise 1.2: Object Oriented Programming

As a first prototype, you try to program a centralized IoT controller, which should control the smart light bulbs. You start with one light bulb and two buttons that can only be temporarily pressed but not switch between two states. In order to avoid spaghetti code, you decide to learn Object Oriented Programming (OOP).

1. What are the four main concepts of object oriented programming?
2. What is the difference between the terms: *class*, *data structure*, and *type*? Give a short definition.
3. What is the difference between *class*, *object*, *instance*, and *reference*? Give a short description and assign these (maybe several) terms to each identifier in the following code:

```
1 class LightSource()  
2 class LED(LightSource) # inherit from LightSource  
3 green_led = LED()  
4 red_led = LED()  
5 power_led = red_led
```

4. Unfortunately, while you were studying OOP concepts, your co-worker (who wants to stay anonymous) has already implemented the first draft over night and pushed it to the git repository. Look at the following bad code and identify how it could be improved with OOP concepts.

Rewrite the code and declare the 3 classes Controller, Light and Button. Either give a high-level code on paper or refactor the full code file, which is uploaded on Moodle (**recommended**).

```
1 def main():  
2     print("Starting_controller...")  
3     init() # controller  
4     init_light_and_buttons()  
5     light_state = OFF # state of the light  
6  
7     print("Entering_main_loop:")  
8     while True:  
9         (b1, b2) = receive_button_states(2)  
10        first_button_state = b1  
11        second_button_state = b2  
12  
13        if first_button_state == PRESSED:  
14            if light_state == OFF:  
15                kitchen_light_on() # power light bulb  
16                light_state = ON  
17            else:  
18                kitchen_light_off()  
19                light_state = OFF  
20  
21        if second_button_state == PRESSED:  
22            if light_state == OFF:  
23                kitchen_light_on()  
24                light_state = ON  
25            else:  
26                kitchen_light_off()  
27                light_state = OFF
```

? Python

Since we will use python for our code examples and tutorials, it might be the perfect opportunity for you to learn Python or improve your Python skills. Take a look at the [Python in 10 minutes tutorial](#) for the basics or [Python Class and Objects](#) for the OOP concepts of Python.