

Assignment issued: Friday, 21 June, 2019
Submission due: Thursday, 4 July, 2019, 23:59 CEST
Presentantion: Friday, 5 July, 2019

Task 1: Terminology of Aspect-Orientation (7 marks)

In your own words and using an example (1 mark), describe the basic concept of aspect-orientation and explain the following terms (1 mark each).

- (a) aspect
- (b) pointcut
- (c) joinpoint
- (d) advice
- (e) inter-type declaration
- (f) quantification

Task 2: Pointcuts in AspectJ (9 marks)

Alongside this assignment, you can find a basic version of the chat application¹. Specify a pointcut in AspectJ to extend the following joinpoints in the provided chat application (1 mark each, except (d) and (g)).

- (a) every execution of method `Client.send(text)`
- (b) every invocation of a method with name `send`
- (c) every execution of a `public` method
- (d) every invocation of a method `send` in class `Client` not originating in the class `Client` itself (2 marks)
- (e) every read-access of the socket on the `Server`
- (f) every instantiation of class `Connection`
- (g) every method invocation inside the method `Server.broadcast()` unless the target is an instance of class `Connection` (2 marks)

Task 3: Advices in AspectJ (Bonus/DE students, 5 marks)

On the right side, a source code snippet is provided with some lines underlined. The underlined lines of code have been added by an aspect. What might this aspect look like?

Write an aspect in AspectJ that transforms the non-underlined lines to the given snippet.

```
1 public class Foo {
2     void main(String parameter, Locker locker) {
3         int lockId = locker.lock();
4         try {
5             int i = parameter.length();
6             firstOperation(i);
7             log(" first Operation executed with parameter " + i);
8             secondOperation();
9         } finally {
10            locker.unlock(lockId);
11        }
12    }
13 }
```

Task 4: Obliviousness Principle (5 marks)

- (a) What is the *obliviousness principle*? (1 mark)
- (b) What are its advantages and disadvantages? (2 marks)
- (c) How can one solve the *fragile pointcut problem*? (2 marks)

Task 5: Implementation in AOP (10 marks marks)

Re-implement the chat application SPL from the previous assignments using aspect-oriented programming with AspectJ (2 marks). The application should have the following features, which can have a simplistic realization: Color, Authentication, Command-line Interface, Logging (2 marks each).

Note: Eclipse versions newer than 4.7 (Kepler) are not supported by AspectJ/AJDT.

Submit your answers (PDF) and implementation (FeaturIDE project) as an archive with name and matriculation number until 4 July, 23:59 CEST to stefan.muehlbauer@uni-weimar.de.

¹https://www.uni-weimar.de/fileadmin/user/fak/medien/professuren/Intelligente_Softwaresysteme/Downloads/Lehre/SPLE19/task_2.zip