Introduction to Software Engineering
Assignment 7

Walter Tichy

Title page: Create a title page with ”Introduction to Software Engineering”, “Assignment 8”, your name, and date of completion.

Problem 1: (2 pt) Explain why Template Method is an instance of “switchless programming”. Use UML and proper English (complete, correct sentences) in your explanation.

Problem 2: (6 pt) This problem is about implementing the design pattern Observer in correct Java. (This problem was an exam question, so try to do it from memory, without looking at the pattern.) Use the conventions for the implementation of UML class diagrams presented in class. Include constructors.

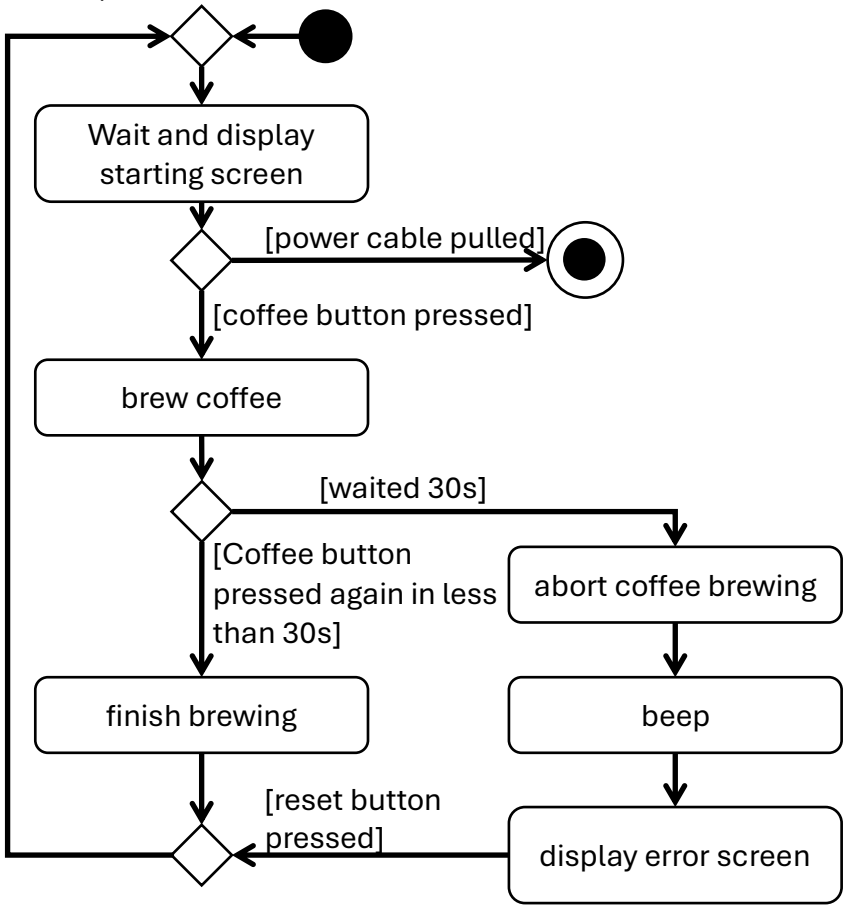
a) (4 pt) Complete the following classes:

```
public abstract class Subject { … }

public class ConcreteSubject extends Subject {
     State subjectState;

     public State getState() {
          return subjectState;
     }
     …
}

public class ConcreteObserver extends Observer {
    State observerState;
   …
}
```

b) (1 pt) The above assumes the *pull* model. Discuss what needs to be changed for the *push* model.

c) (1 pt) Discuss the advantages and disadvantages of the *push* model vs. the *pull* model.

Problem 3: (4pt) The coffee machine of your team is not working right. The main problem seems to be that one must press the coffee button twice, otherwise the brewing process stops. Your boss who needs coffee to work has drawn an activity diagram and now asks you to fix the machine. To get a better view of what happens when, you decide to figure out what the states of the machine are. The activity diagram is below. (SS19 ü4)



Convert the UML activity diagram above into a UML state chart. Model the states of the machine as accurately as possible. Your UML state chart should have as few states as possible. Provide the state transitions with events and operations and use entry, exit and continuous actions as well as conditions if necessary.

Problem 4: (6 pt) Convert the state chart you produced in the previous problem into a UML class diagram according to the state machine design pattern presented in class.