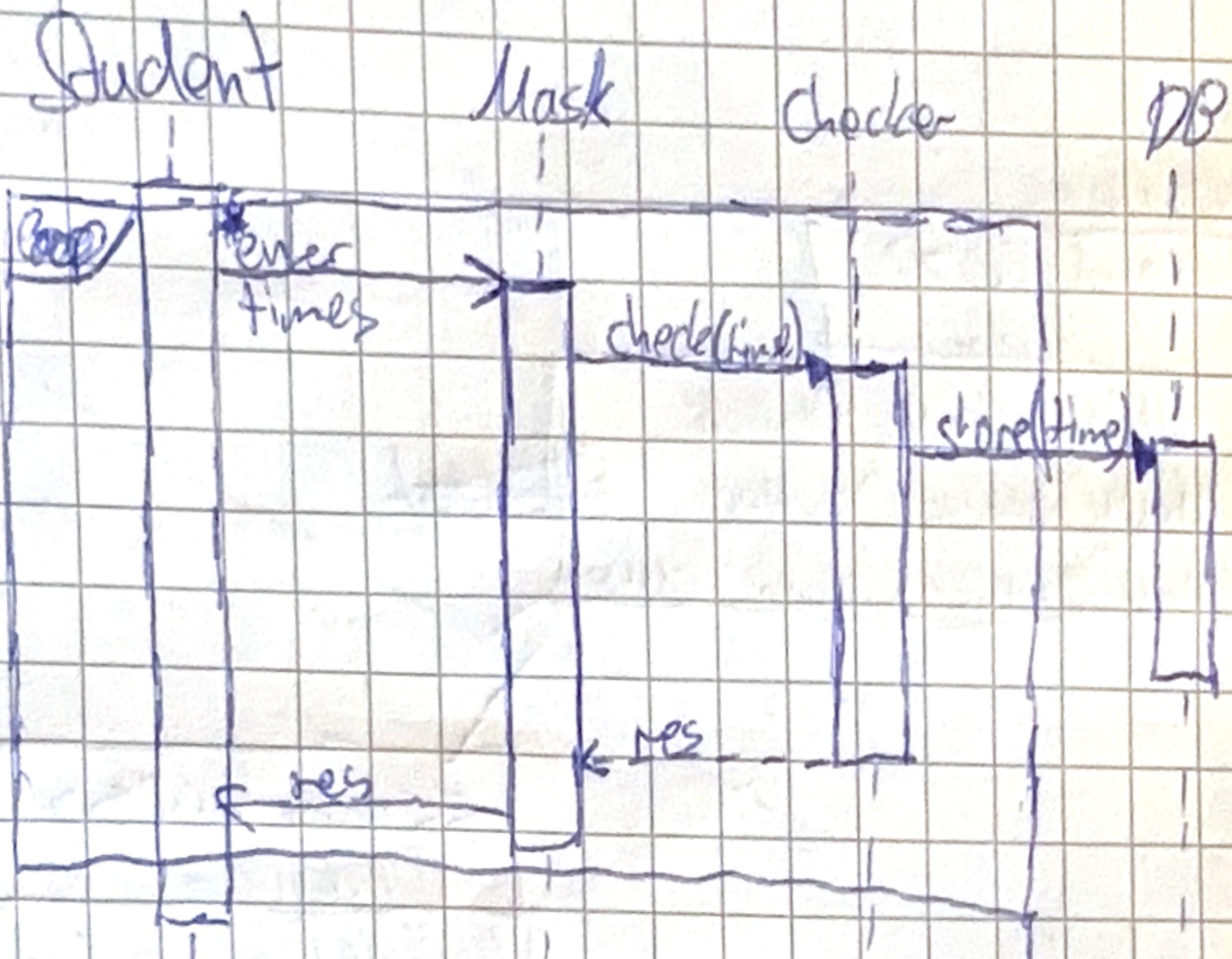


Mr. 1



if condition: conflict with times

Mr. 2

a) context CourseManager :: addCourse(courseName: String)

pre: courseName != null and courses \rightarrow !includes(courseName)

context CourseManager :: addCourse(courseName: String)

post: courses \rightarrow includes(courseName) and courses \rightarrow size = courses @ pre \rightarrow size + 1

context CourseManager

inv: courses \rightarrow forAll(c | c) = null and courses \rightarrow isUnique(c | c)

6)

SRP: - Problem: Speicherung und Methoden in einer Klasse

- Lösung: Speicherung des Array/ide im Separaten Ort

OCP: - Problem: Sollten neue Anforderungen wie für z.B. Formate kommen, so müsste man bestehende code modifizieren

- Lösung: Strategy pattern nutzen um Anforderungen hinzufügen ohne die Änderung der Klassen

LSP: - Es entstehen keine Probleme da nicht mit Subtypen gearbeitet wird

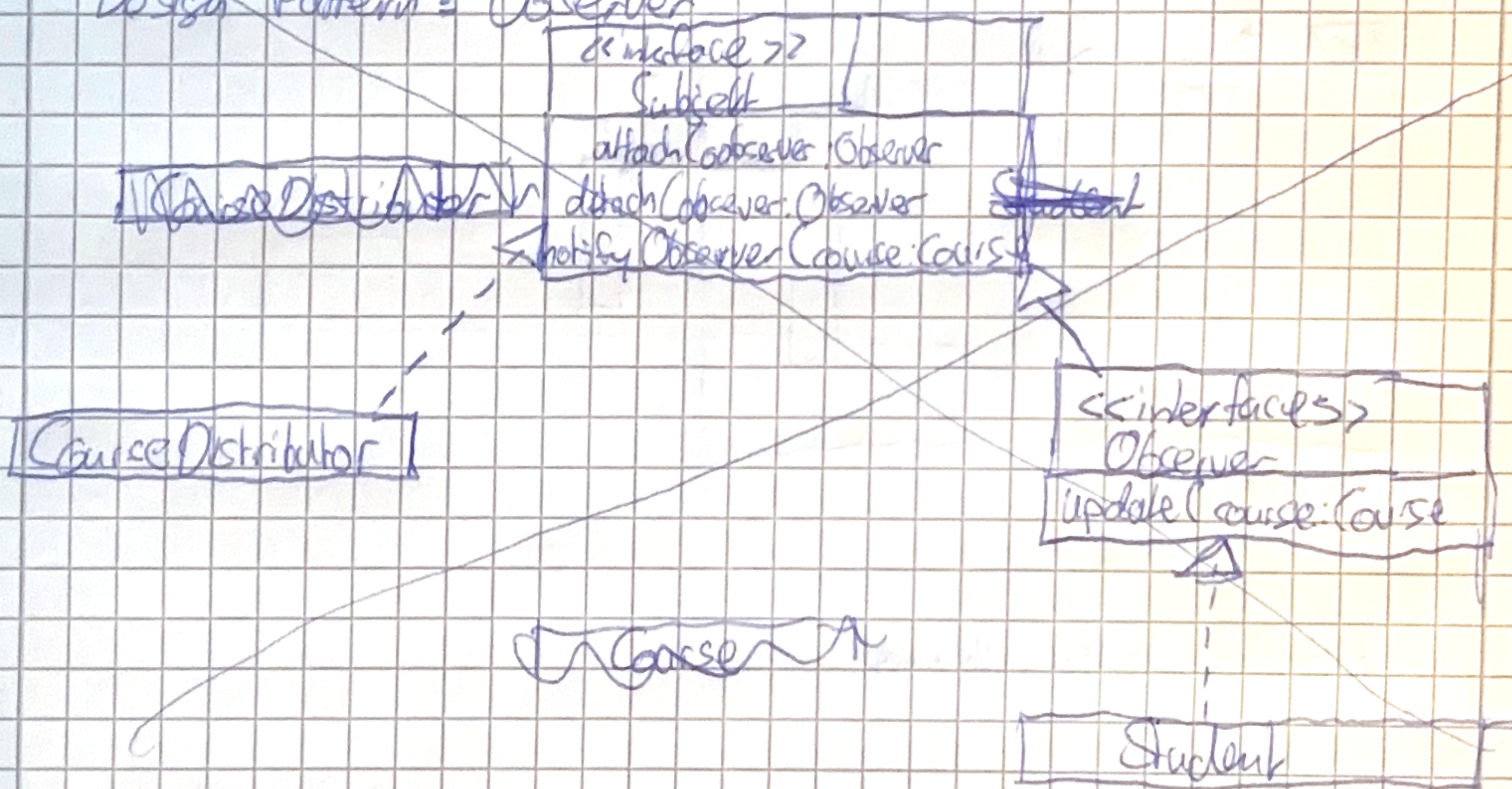
ISP: - Es wird auch nicht mit Interfaces gearbeitet somit liegt kein Problem vor

DIP: - Problem: Course Manager ist direkt abhängig von der konkreten Implementierung der Array/ide. Sollte man diese Speicherung ändern muss man den Ort der

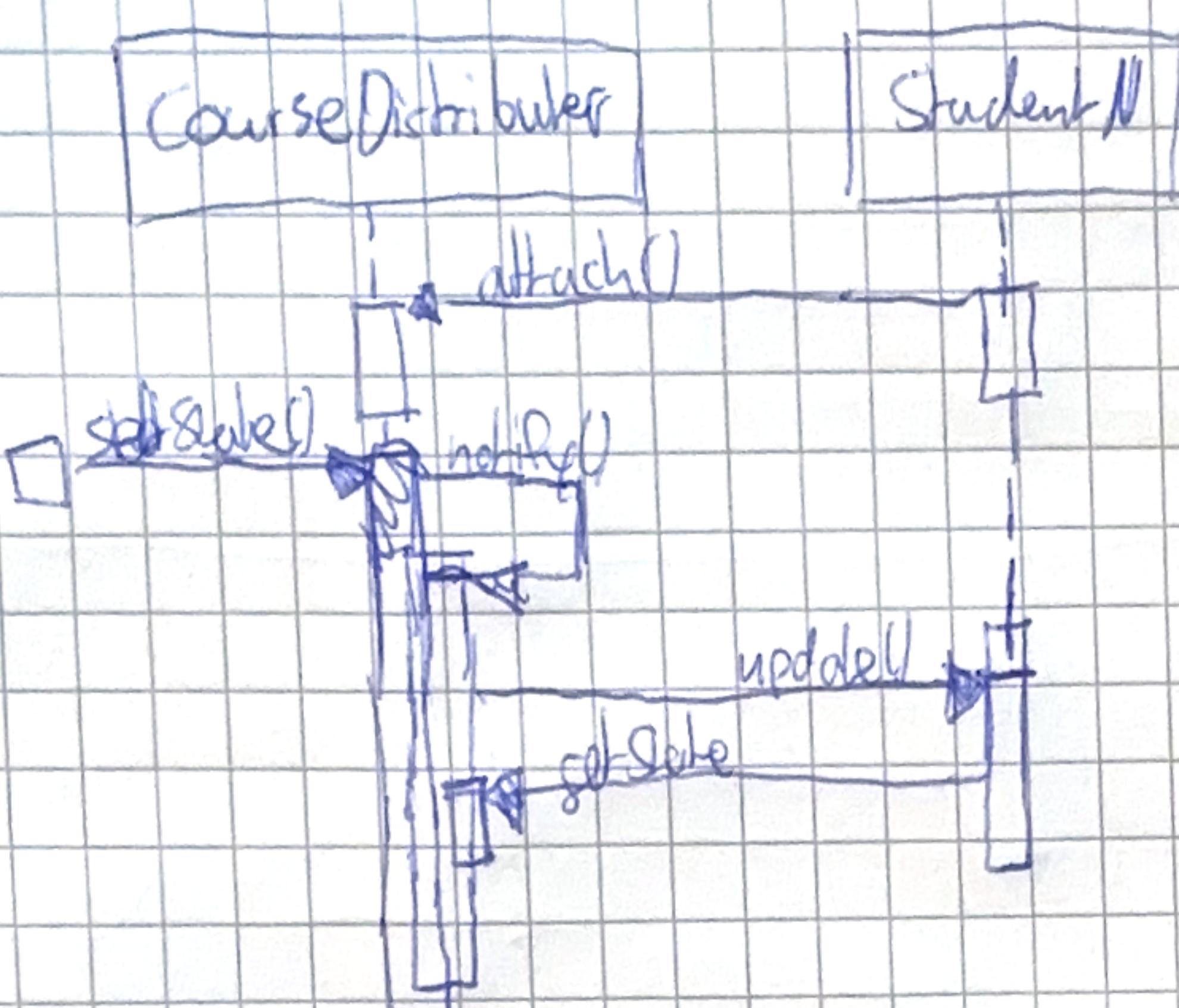
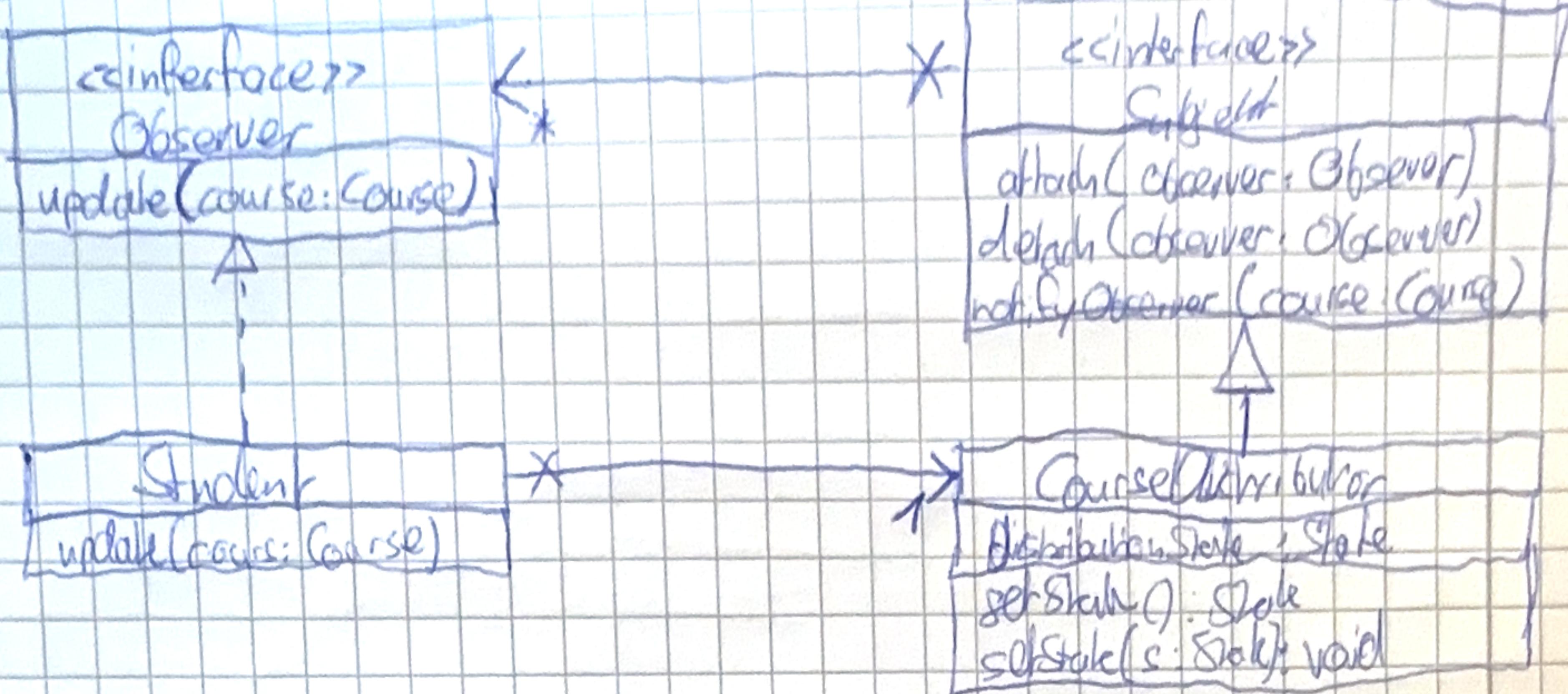
viel ändern
- Lösung: Lieb interface benutzen

Nr. 3

Design Pattern: Observer



Design Pattern: Observer



Mr. 4

