OBS!

```
 <u></u> Means underlined
```

FD Diagram

```
D(studentIdnr, studentName, login, branchName, programName, courseCode, courseName, credits, departmentName, capacity, classification, grade, position)

login -> studentIdnr

studentIdnr -> studentName
studentIdnr -> programName
studentIdnr -> branchName
studentIdnr -> login

courseCode -> courseName
courseCode -> credits
courseCode -> departmentName
courseCode -> capacity

position courseCode -> studentIdnr
studentIdnr courseCode -> grade
studentIdnr courseCode -> position
```

BCNF

After utilizing the BCNF algorithm we get 3 separate tables

```
 1. Students
 studentIdnr, studentName, login, programName, branchName
Keys: studentIdnr or login
with FDs
studentIdnr -> login, studentName, programName, branchName
login -> studentIdnr

2. Courses
courseCode, courseName, credits, departmentName, capacity
Keys: courseCode
with FDs
courseCode -> courseName, credits, departmentName, capacity

3. Taken and waitinglist
studentIdnr, courseCode, grade, position
Keys: courseCode stdentIdnr or courseCode position

courseCode studentIdnr -> position grade
courseCode position -> studentIdnr

4. Classifications and Registered
studentIdnr courseCode classification
```

4NF

To violate the fourth normal form a table should have a multivalued dependency, which means that for a table with at least 3 columns there should be 2 that are independent for every row with a column of value A more than one value of of a column B exists

We can see this in the Classifications and Registered table, for every course there can be multiple classifications and multiple students. The student value and classification of the course are independant. Therefore we should split this table in two

```
classified(<u>course</u>, <u>classification</u>)
    Courses -> course.code
    Classifications -> classification.name

registered(<u>course</u>, <u>student</u>)
    course -> Courses.code
    student -> Students.idnr
```