**PA6 Grading Rubric – Room Scheduler**

**Point deductions below are given in percentages. This is a 300-point assignment, so the point deductions will be 3 times the numbers listed below.**

Late policy of 4% per date late and using good object-oriented coding still apply.

Copying code from the internet does not count as writing your own code. Do not do this.

A good way to check your code is to use the test script provided.

There are four parts to this part of the project:

Adding a faculty, adding a reservation for a faculty, room, and date, displaying the faculty reservations for a specific date, and displaying the waitlist.

Preferably, the code you submit will be only for part 1, but if your code includes functionality for the second part of the project, that’s ok too; we will only grade for the functionality for part 1 (so you need to make sure all parts for part 1 still work).

Database and project considerations:

* When you submit your project, you MUST submit both your code and your database (-5 if you fail to submit your database when you submit your code. We need it to grade your assignment.)
* You need to close your project before you zip the database otherwise the graders will receive errors when they grade it.
* You must submit the zip of your entire NetBeans project folder instead of just the .java files
* Your database should have the username and password of ‘java’ and ‘java’ as per the directions
* Your database must be preloaded with several rooms and dates (-5 for an empty database)
* Your preloaded database must have empty reservations/waitlist tables (-5 for not following these directions)
* Every room should be able to be reserved on every date. In other words, you should not have rooms associated with specific dates in your database unless it is through the reservations table. If you find that confusing or have questions about that, talk to Al or a TA. (-10)
* You should have the following tables in your database with the appropriate columns: Rooms, faculty, dates, reservations, waitlist (can optionally be combined with the reservations table) (-10 for each table)
  + You should also have the appropriate primary keys. For example, you should not be able to add two people with the same name to your Faculty table
* You must use SQL Prepare Statements (-10) when updating the database

Functionality:

* Correctly adding a faculty to the database (-10)
* Correctly adding a reservation to the database (-10)
* Correctly checking status by date (-10)
  + For status by date, you should display the name and room of the faculty that have a room reserved on that date.
* Checking waitlist (-10)
  + Note: You should only display the waitlist here.
  + For waitlist, you should display the faculty name, the date, and the number of seats that the faculty is waitlisted for.
* If a room is not available, the faculty reservation should be put on the waitlist (-10)
* If rooms are available, the faculty will be assigned the smallest room for the requested date (-10)
* The priority order of the waitlist must be maintained (-5)
  + It is strongly recommended that you use Timestamp to keep track of a faculty’s position on the waitlist

GUI:

* Although the tab layout is recommended, it is not required. If you designed your GUI using a different type of layout, that’s totally fine – just make sure you have all the required pieces of functionality and show all the appropriate information.
* You should have combo boxes to select the faculty, and date when making a reservation or checking the status (-2 for each combo box missing)
* The combo boxes should be updated automatically when additional data is added to the database (-3)
* When a command is performed, the results of that command should be displayed to the user without the user needing to check the status to see what was done. For example, if you add a faculty to the database you should display the name of the faculty that was added and if the faculty was added to the database successfully. Likewise, after adding a reservation, you should display which room was reserved on which room on what date. If all rooms are reserved, you must display information that the faculty reservation was added to the waitlist
* How you want to display status information is up to you, but it must include all necessary information and be readable to the grader.