Dear students,

Below are the most frequently asked questions of this week's lab (week 8):

Question 1: For the consistency check, should we compare rows from *S* or rows from *S* against rows from *SA*?
Answer:
If you follow the definition from Angluin's and Vaandrager's papers, the consistency check is only done by comparing rows within *S*.

Question 2: We have implemented the check for closed and consistency but our code never exits the while loop containing the checks. Is this supposed to happen?
Answer:
Finding a consistent and closed table could take some time but it shouldn't take too long. If you don't get out of the while loop containing the checks, then your implementation of the close and consistency check might not be (entirely) correct. Double-check your implementation with the help of the definition listed in the papers

Question 3: In the performance statistics of Lab 4, a depth of 3 is used. What is this depth for the W-method?
Answer:
The depth corresponds to the "*w*" parameter that is used in the constructor of `WMethodEquivalenceChecker.java` or the "*k*" variable that is used in the lecture.

Question 4: In the consistency check, we would have to go through all one-letter extensions for rows *s* and *s'*. How do we get the one-letter extensions, do we have to create them ourselves?
Answer:
Within the framework that we use for the lab, we automatically add the rows for one-letter extensions in the observation table for each element that is added to *S*. You can thus query the row for the one-letter extension using the table e.g. as shown in the example you can get the row for *sa* by doing the following: `ArrayList<String> rowSA = table.get(sa);`, where *sa* is the one letter extension that you create using a row from *S* and an input symbol *a*.