*Q1. Can we use non github repositories?*

A1. Yes, gitlab and bitbucket could be viable alternatives. But kindly do not use something quite uncommon that might require us to create our own account.

*Q2. For fuzzing/testing other implementations (not written by me), can we change their code?*

A2. The general answer is NO (for obvious reasons). However, in certain cases, such changes might be acceptable. In particular, such changes need to be solely for the purpose of testing (e.g., getting certain coverage via print statements) and must not affect the “core” functionality of the code being tested. In your Week 13 report, you must document all the changes made to the code being tested, why they were made and why the changes are rational (in line with the argument that they do not change the core functionality). Note that when you test the implementation written by others, your objective is not to fix their code.

*Q3. Do we get additional marks if I find bugs in third-party implementation?*

A3. Yes, you get 5 marks per **unique** bug found. Note that a bug can be manifested by many inputs. You will certainly not get additional marks for each input manifesting the same bug. 5 marks are given for each *unique bug.* It is your job to convince why each reported bug is unique. When in doubt, consult with Sudipta.

*Q4. Can we consult with the author of third-party code being tested to understand their code?*

A4. Short answer is NO. However, despite providing all guidelines, we understand some code may not be providing a proper README file. In this case, it is OK to have some consultation with the authors. Please try your best to minimize such communication.

*Q5. What if someone gets a really weak piece of code allowing him/her to find many bugs?*

A5. That is why everyone’s job is to make sure their code is well tested and well documented. It is some sort of game where we are hoping to reach a stable point. Additionally, you are certainly not penalized for not finding bugs in other’s code.

*Q6. I am from one of the SES groups and I do not have three other group members. How do I get three other implementations?*

A6. No worries. We will arrange another implementation from a different SES group.

*Q7. How do I identify myself in the github repository?*

A7. Make sure to include your name and student ID in the beginning of the README file.

*Q8. if the two CSV files contain the same unique combination (same id, balance, currency, etc) but are stored on different rows in the two files, do we consider them to be mismatched?*

A8. This is not a mismatch. The same combination may be located in different lines in the two CSV files.

*Q9. Will all inputs only contain the types of data as shown in the use cases? Ie. should we expect to receive differently labelled data such as maybe {student name, student id, grades}, or is considering the example case sufficient?*

A9. Of course, the provided samples are just examples. You may expect different types of data with different column labels (e.g., student ID, grades). However, note that you may not need to handle all cases by the first week. For example, in the first week, you may handle only data in a given format (e.g., used in the example), and error out on if the provided data does not meet your expected format. That is why we have a testing and possible refinement stage, where you can extend your program with features/fix buggy features.

*Q10. If one of the files has an entry with an ID that does not have a corresponding entry in the other file, is an exception expected? Also, what if there are multiple entries in a file that are identical?*

A10. For the first case, it is an exception. Multiple entries, if identical, will not be counted as exceptions (if they match with the corresponding content in the other file). For example, if one file has <x,y,z> and the other file has <x,y,z> three times, and we take a combination of the first column and second column for comparison, then there is no mismatch even though “z” appears thrice in one file as compared to the other for the combination <x,y>. However, if one file has <x,y,z> and another has <x,y,z1>, <x,y,z2>, <x,y,z3> then there are three exceptions for the combination <x,y>.

*Q12. Do we assume that both CSV files share the same set of columns i.e. CSV1 = CSV2 = {A, B, C} and that we use all of these columns as the unique combination? Or do we have to account for the case where we have to identify the unique combination i.e. CSV1 = {A, B, C, D, F}, CSV2 = {A, B, C, E, G}. Therefore, the unique combination is {A, B, C} and we only compare the values {A, B, C}. However, this may not make sense even if the records share the same values for {A, B, C}. The extra columns invalidate the entire records due to missing columns or this can be seen as a feature similar to inner join. So, do we terminate the program or proceed with the comparison?*

A12. <Answer>

*Q13. Do we assume that both CSV files have their columns in the CSV file? What do we do if both files have no columns? Do we still perform a comparison or should we declare an exception? This can be interpreted in 2 ways. The user can argue that the values are ordered in the same way for both files and hence a comparison can be made. However, there is no way for the program to know this. So, should we provide this as a feature or should we just immediately declare an exception and terminate the program*

A13. <Answer>

*Q14. For example if file1.csv has 100 rows and file2.csv has 100 different rows, does this mean we will end up with 100 exceptions as the rows in file1.csv are not found in file2.csv or do we end up with 200 exceptions?*

A14. <Answer>

*Q15. <Question>*

A15. <Answer>