*Critical Reflection on* “*Formative Assessment of Programming Language Learning based on Peer Code Review: Implementation and Experience Report*”, *By* *Qing Sun, Ji Wu\*, Wenge Rong, and Wenbo Liu*.

In this article, they designed a formative peer assessment approach based on the peer code review (PCR) model and implemented it in a programming learning course. Students anonymously reviewed the program/code written by their peers against the pre-agreed criteria. This aspect of an anonymity in the review process helped reluctant and hesitant students to voice out their knowledge and provide feedback of their peer’s work. Both the author (students who wrote the code) and the reviewers are not left embarrassed of their work (mistakes), promoting an ethical assessment practice. Specifically, I did not intend to use peer assessment but somehow unknowingly, I have implemented peer-assessment in my module whereby the students were asked to give reviews of the work shown by their friends. For instance, asking the students whether the program/code written by their friends will give the intended output or not in the classroom. Since I have allowed the open code review in the classroom, I feel I have not incorporated the anonymity of the reviewer which might have led to anger and frustration by the author.

An arbitration for non-consensus on reported bugs/code were also implemented in the peer assessment where if the author disagrees with the reported bug, they can initiate the appeal to the respective reviewer. The solving of non-consensus on reported bugs is very important to the reliability of peer assessment as it might lead to an argument between the reviewer and the authors, ultimately causing distrust or chaos among the students. By doing so, it further promotes positive outcome in the process of the peer assessment and fair assessment conducts. In my class, I did similar kind of implementation whereby if the author is not satisfied with the feedback pointed out by the reviewer, he/she can ask for justification/clarification. Mutually, they tried to solve their disagreement and if both are not sure of the answer, I as a teacher intervene the process to evaluate the effectiveness and provide relevant feedback. This aspect helped all the involved students and teacher to enhance the learning.

They also implemented the dynamic reviewer appointment strategy according to student’s programming performance which means the higher-ranked student review programs written by lower-ranked students. Teachers review the program written by the highest-ranked students while the students of equal rank review each other’s work. This allowed balanced assessment for diverse types of learners which allowed them to have the maximum exercise of each student’s ability. As a teacher, I didn’t implement such a remarkable strategy where the abilities of the students are studied and based upon their ability, we can initiate the peer assessments. I would randomly pick the name or roll number of the student to give feedback of their friend’s code/program irrespective of their ability. In doing so, I haven’t considered the weaker students who otherwise might not be confident to give feedback to their peer’s work.