In this study, the author examines a student drop-out ratio on a novice programming course in Tampere University or Technology. Drop-out rates have been high due to somewhat large final assignment on the course: students have not been able to accomplish it in an acceptable manner. The phenomenon happens due to too late start on the project. To solve the problem the author’s working team has split the final assignment into four separate phases to force to students to start working in time and additionally to transform the one large assignment in to more concrete divisions.

Course concerned here is a second programming course for 1st year CS major students, who will learn the basics of object-oriented programming using C++. Course tasks consist of four short programming assignments and one more in-depth one covering the whole course substance. This task is supposed to be accomplished individually. Approximately 300 students participate in the course and during recent years, almost half of them have failed mostly due to problems in accomplishing the in-depth assignment. The scale of this in-depth assignment is substantially larger than in the first programming course, where large assignment consists of approximately 200-400 lines of code, whereas in this second programming course the final assignment consists of over 2000 lines of code. Although students are heavily advised during the course, a great amount of them enter to this assignment too late to accomplish it in time.

The author’s working group has altered the course practices to force the students to start working earlier. The guidance has also been changed in the way that the students are encouraged to approach this problem by first sketching the big entities rather than starting with a line-by-line viewpoint. This process is described to be iterative. This study aims on measuring how well this alteration process of the final assignment has succeeded.

Iterative assignment process
In this final assignment students are supposed to implement an ASCII-based battleship game. Students are to create classes out of ships and players and implement modules for the user interface (shell) and for file parsing. Assignment was divided into a design phase, four actual implementation phases and the final submission. According the implementation phases, the minimum requirement was to attend every other of them.

In the design phase, students were expected to design and document entities for the program (classes, modules and interfaces). Personal feedback was given from the design. Due to students’ design difficulties, the feedback session was in practice mostly guiding the students to the right track with their implementation. Phase one was to implement the parsing and checking of the beginning arguments. (Easy start.) Phase two was to implement the user interface, a text shell. (More complex, but mostly repetition of old assignments.) Phase three was to create the basic initialization related functionality or one’s program: file parsing and initialization of objects to
match the configuration file. Phase four was to define the game logic. Final submission is left for an automatic grading system to test the program functionality. After passing the automatic assessment it’s accepted for manual grading process.

**Research questions**
- To measure the drop-out ratio
- The phases of drop-outs
- Time used, perception of workload
- Students assessment of iterative assignment process and it’s usefulness
- The quality of submitted works compared to earlier years’ works

Was this work beneficial or not?
Trying to find reasons for students’ perception of the assignment being difficult to accomplish. Is this related to topics or scale of this assignment or something else that could possible be fixed by altering the assignment and/or it’s guidance?

Methodology: mixed methods, emphasize on quantitative methods

**Data collection**

**Quantitative:**
- The amount of assignment starters (= the amount of first submissions)
- The amount of submissions
- The timing of submissions
- Mandatory surveys connected to submissions
  - How much time did You approximately use for this phase?
  - How easily did You manage through this phase?
  - How clear is the implementation of the next phase for You?
- Mandatory survey connected to final submission
  - Did You find this iterative assignment more laborious than the assignment would have been without the phases?
  - Did the phases help You in accomplishing the assignment?

**Qualitative:**
- Open feedback form connected to the documentation related to final submission

**Discussion of the relevance of this project from scientific view on theme ‘Quality and trustworthiness in CER’**

This seems to be a case study on a course development process. The result will be some kind of comparative analysis if this development process has changed student behavior compared to earlier or not. This earlier product seems to be a simple ‘an assignment and return’-policy. However it has been mentioned that also earlier version of this final course assignment included a lot of guidance. What kind of was the earlier guidance? Can this earlier guidance be characterized as an assignment
product or not? What are the strict differences between the earlier guidance compared to this ‘iterative assignment tool’?

Is the difference between developed and earlier guidance merely the mandatory nature of this new guidance? (I.e. was the earlier guidance look-a-like with the difference that it did not include mandatory submissions? Or was it something else?)

Towards what we are comparing this new final assignment product?

First important issue is the data this developed final assignment is compared at. Is there detailed data available from the traditional course concept available? Is it measurable? Is it comparable?

If the old data is sufficient and comparable, the quantitative numerical analysis seems to be valid. However, do these figures tell us anything really?

Can this research question be simplified into form: if we provide students with better guidance or increase constraints, will they perform better?

As the author depicts, with the submission survey, we have the problem that the students might not think in-depth enough about the questions they are asked while submitting the answers and this way they might end up with too superficial data. Additionally as the author describes, the problem might be that since the questionnaire is related to the submission, the students might not consider it anonymous and they might lie to give better picture of themselves. Queries are not anonymous. (To compare the time load to the achieved grade.)

Since this is a case study, generalization seems not to be an issue here. The results are to be considered as indicative.

However, this seems to be a very interesting development project.