Respondent: **Simo Antikainen** Submitted on: Wednesday, 6 December 2017, 9:27 PM

# Mid-term review of C++ project

Name of the project group evaluated

q-learning-9

C1.1: The implementation corresponds to the selected topic and scope. The extent of project is large enough to accommodate work for everyone (2 p)

Everything is good. Project is large enough. 2/2p.

C1.2: The class structure, information hiding and modularization is appropriate, and it is explained and justified in documentation. The file structure corresponds to the class structure (2 p)

**Everything is good. Class structures are appropriate and they are used well. Classes are good explained  2/2p**

C1.3: Use of at least one external library (in addition to C++ standard library). Comment the appropriateness of libraries and their use. (2 p)

Box2D and Google test.  Did not find comments of their use. 2p

C2.1: Git is used appropriately (e.g., commits are logical and frequent enough, commit logs are descriptive) (2 p)

Git commits have small granularity and are used to commit functional features into branches. Commit logs are descriptive enough taking into account the small size of the commits. Graphs and charts show that the development of project is active and commits are frequent. Different branches have been used to separate large features. (1p)

C2.2: Make or Cmake (recommended) is used appropriately. The software should build easily using these tools without additional tricks. Nevertheless, instructions for building the project should be provided (1 p)

**Project builds as of writing this using Makefile with Cmake in the root .There exist clear build instructions in the root folder for running Q-learner ands its tests. (1p)**

C2.3: Work is distributed and organised well, everyone has a relevant role that matches his/her skills and contributes project (the distribution of roles needs to be described) (1 p)

**Distribution of roles has been described well in the project plan and followed thoroughly; Mikael and Anssi have built the learner in the master branch, and Visa and Lassi have built graphics and simulation in a separate Box2D_old branch. Outside of this division: Mikael and Anssi have been building tests (1p)**

C2.4: Issue tracker is used appropriately to assign new features and bug fixes (1 p)

**Gitlab issue tracker has not been visibly used in the issues tab. Just trying to use it would go a long way. (0p)**

C2.5: Testing and quality assurance is appropriately done and documented. There should be a systematic method to ensure functionality (unit tests, valgrind for memory safety, separate test software and/or something else.) (1 p)

Test_source.cpp file contains a remarkably extensive suite of tests concerning all the essential classes of the program; including the parts essential to physical interaction and displaying movement. These tests have been organised well into functional units which

facilitate fast and easy bug fixing.  Test have been built using industry proven googleTest library, but valgrind has not been included as part of Makefile. (1p)

C3.1: C++ containers are used appropriately (including appropriate use of iterators), and justified (e.g., why certain type of container over another) (2 p)

The project mostly uses vectors for and maps to store data. The use of these is justified as insertion is mainly done though the end and is iterated “forwards”.

(2 p)

C3.2: Smart pointers are used in memory management, describe how (1 p)

I didn’t see any code using a smart pointer.

(0p)

C3.3: C++ exception handling is used appropriately, describe how (1 p)

Error handling is used, but error classes were absent. Usage of these classes would clarify the error handling a bit.

(0.5p)

C3.4: Rule of three / rule of five is followed, describe how (1 p)

No class explicitly defines a destructor, copy constructor or copy assignment operator, so there is no need to follow the rule.

(1 p)

C3.5: Dynamic binding and virtual classes/functions are used, describe how (1 p)

Didn’t see templates being used. Virtual classes are used to implement different actors and sensors for the learner.

(0.5 p)

Other comments and feedback to the evaluated project group.

If you did this review together with (some of) your group members, list the names of the group members here. Everyone needs to turn in a review, either separately or as a group.

Simo Antikainen

Aleksi Korsman

Santeri Porrasmaa

Andrei Spelman