ELEC-A7150 - C++ Programming, 12.09.2017-15.12.2017

Preview

View All Responses

Non-respondents

Summary

List of responses

Individual responses

<< First Respondent | **<** Previous | **112 / 116** | Next **>** | Last Respondent **>>**

<<< List of responses | Print this Response

Respondent: **Antti Virtanen** Submitted on: Monday, 18 December 2017, 11:32 AM

Mid-term review of C++ project

Name of the project group evaluated

settlers-2

This is a project self-evaluation

☐ Yes

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)

2p The project is large enough and corresponds to the topic. The topic is broad and we realize that it has been limited in scope.

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)

2p Every object has its own class with header and .cpp files. The class structure is explained in the documentation. File structure corresponds to the class structure, however files could have been more organized, for example images in their own folder.

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)

2p The graphics are done with Qt library. This is an appropriate choice.

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

2p Git is used appropriately. The commits are frequent and logs descriptive. Branches are used well and Git graph looks nice.

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)

1p Cmake is used appropriately and the program builds simply with the make command. Instructions were given in the documentation.

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)

0.5p Work is distributed and organized mostly well. One participant apparently had less work than others.

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

1p Issues are used to track bugs, and features to be included.

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)

1p Testing was documented and even though there were memory leaks to be found with running Valgrind they were justified.

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)

1.5p The program has iterators and vectors as containers. They are not explained in words in the documentation but they are justified.

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

0p We could not find any smart pointers whatsoever.

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

0.5p There are couple of try - catch structures used in appropriate places. The catch statements catch everything which is not very good practice.

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

0p Destructors are implemented but copy/move/assignment constructors are missing i.e. rule of three is not followed.

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

1p Virtual methods are used in the building class.

Other comments and feedback to the evaluated project group.

The game looks nice but was a bit hard to play since nothing in the UI showed what was selected and clicking didn't seem to work intuitively.

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.

Antti Virtanen, Marton Gunyho, Jani Taskinen, Jani Mäkinen