Concordia University
Department of Computer Science
and Software Engineering

Advanced Program Design with C++
COMP 345 --- Fall 2014

Project Build 1 Grading

1. First Incremental Code Build Description

You must deliver an operational version demonstrating some capacity of your system. This is about demonstrating that the code build is effectively aimed at solving specific project problems or completely implementing specific system features. The code build must not be just a "portion of the final project", but rather be something useful with a purpose on its own, that can be demonstrated by its operational usage.

The presentation should be organized as follows:

1. Brief presentation of the Problem Analysis, Design, and Use of Engineering Tools as listed below under "Graduate attributes—skills"
2. Demonstration of the functional requirements as listed below under "Functional Requirements".

You are graded according to how effectively you can demonstrate that the features are implemented. If you cannot really demonstrate the features through execution, you will have to prove that the features are implemented by explaining how your code implements the features, in which case you may lose some marks.

During your presentation, you have to demonstrate that you are well prepared for the presentation, and that you can easily provide clear explanations as questions are asked about your understanding of the problem being solved, the structure and functioning of your code, as well as your use of tools.

2. Team Identification

Team	Evaluator	Signature	Date	Time

3. Grading

Presentation		**6**
Effectiveness, structure and demonstrated preparation of the presentation		3
Knowledge of code base/clarity of explanations		3
Functional Requirements		**32**
Map creation and editing		**14**
User-driven interactive creation of a map as a grid of user-defined dimension		2
User-driven allocation of grid elements such as scenery, path, entry point and exit point.		3
Saving a map to a file		3
Loading a map from an existing file, then editing the map		3
Verification of map correctness before saving (at least 3 types of incorrect maps)		3
Game play		**18**
Game starts by selecting a saved map, then loads the map		1
User-driven placing of towers on the map, following the game's restrictions		2
Implementation of currency and cost to buy or sell a tower		2
Implementation of towers' level-dependent characteristics such as level, cost to increase level, refund rate, range, power, rate of fire, special effects, etc.		3
Implementation of critters with proper characteristics that are shown to move on the map from the starting point to the ending point		3
Tower inspection window that shows its current characteristics		3
Tower inspection window allows to sell the tower		2
Tower inspection window allows to increase the level of a tower, changing its characteristics		2
Graduate attributes—skills		**12**
Problem analysis—demonstration of proper analysis of the problem to be solved and justified adoption of a particular solution		4
Design—description of the rationale and structure of the architectural design and detailed design		4
Use of Engineering tools—justified adoption and proper use of particular tools for the implementation (e.g. compiler, IDE, libraries, project management tools, etc)		4
Total		**50**