Level 1

You may now pursue to the level 1 of the project.

* Board
  The game is played on a series of dungeon levels which are simple square grids. The grid cells may be empty, walls, or floor. Empty cells must never be next to floor cells. Maps can be loaded from and save to a file.

*** Graphics
  The graphics can either be 2D or text based. There is a window where the board is displayed.

* Controls
  The player is controlled with zqsd and can move through floor cells, but should be blocked by walls.

* Monsters
  Monsters appear on the map, and move along some predefined path.

** Fighting
  The hero can attack monsters, and kill them.

* Experience
  The hero can progress and level up doing more and more damages the more monsters he kills.

** Intelligent Monsters
  Monsters will go towards the hero, without getting stuck. In simple situations without other moving entities, an optimal path is used.

Dungeon Battle
A rogue-like game

4 people

Project Description

Dungeon crawls, or more specifically rogue-likes are turn-based computer games in which a single player evolves through a procedurally generated dungeon, fighting creatures, finding various objects, acquiring experience. The final objective may be to reach the top (or bottom, if the dungeon is a cave) of the dungeon and come back, or to find a special object. Death is typically permanent, and plays can be very short, especially for beginners.

Skills

Scenario

Real Time programming

GUI

(*)[The skill scale is from 0 (Fundamental Awareness) to 6 (Expert).]
Dungeon Battle
A rogue-like game
4 people

Project Description
Dungeon crawls, or more specifically rogue-likes are turn-based computer games in which a single player evolves through a procedurally generated dungeon, fighting creatures, finding various objects, acquiring experience. The final objective may be to reach the top (or bottom, if the dungeon is a cave) of the dungeon and come back, or to find a special object. Death is typically permanent, and plays can be very short, especially for beginners.

Skills
Scenario
Real Time programming
GUI

(*)[The skill scale is from 0 (Fundamental Awareness) to 6 (Expert).]

Level 2
Level 1 must be unlocked to read this section

* Developper Documentation
  Required for lvl 2 validation
  Document your project (not necessarily only in the source code) so that a newcoming developper could understand and contribute to the code.

* Release
  Required for lvl 2 validation
  Produce a release as a source archive or git tag. The release files should have up-to-date README and INSTALL files and more generally allow anyone to deploy the application.

* Test Pathfinding
  Test pathfinding on basic examples.

* Test Game mechanics
  Test movements, collision, death...

* Test Map generation
  Generate images of complete level generations, that can be bulked analyzed by a human to judge the effectiveness of the generation. Check that all levels are connex.

** Friendly NPCs
  Some NPCs should be friendly, e.g. a dog that follows the character and attacks nearby enemy NPCs.

* Body snatch
  [optional]
  The player can acquire a spell by picking up a special item. The spell allows the player (once) to exchange his character with another one. After the exchange the player controls the other character and has his characteristics, and conversely.

** Lighting
  All characters should have a limited field of vision. A decent lighting algorithm should be used to determine it. It should be developed in a TDD way.

** Dynamic world generation
  Upon entering a level, the complete world should be generated.

* Memory profiling
  [optional]
  Profile the memory usage of the application, on a specially created game instance that allocates a lot. Fix what needs fixing until a reasonable result is obtained. Demo the evaluation and fixes.

Dungeon Battle
A rogue-like game

(*)[The skill scale is from 0 (Fundamental Awareness) to 6 (Expert).]