Game Developer test for King.com (C++)

Purpose

The goal is to provide a relevant, fun and inspirational foundation for a technical conversation for an upcoming interview. The submitted code itself together with an interview will provide the interviewer with an understanding of the applicants coding style and skills.

By no means is the goal to get a solution covering all special cases in a 100% robust way. Requiring this would be naive. At least the functions described shall be error free when used correctly, how everything else is handled is subject to creativity and ambition.

We will pay attention to code structure and understanding of the problem, as well as the discussion around design and choice of data structures. We will also pay attention to the “feeling of the game”, looking at animations, transitions and other things that enhances the players experience.

Game Concept

- The game consists of a 8x8 grid with different colored objects
- The objects can swap place as long as they are positioned next to each other and on the same row or column (up, down, left, right)
- If a switch result in three or more object of the same color in a row or column, these objects are removed from the game
- If a switch does not result in a match, the objects are automatically switched back
- When objects are removed from the game, the game board collapses so that object positioned above the removed ones falls down to take their place and new objects falls in from the top to fill the board.

Facts & Features

- 1 minute long

- 5 colours

- Drag or click objects to swap

- Use a game like Midas Miner for reference (www.king.com)

Tools & Frameworks

- You should develop the game in C++

- You shall use libSDL as library to get you to a good game (http://www.libsdl.org/)

- If you believe that you can do a better game with your ”own” engine – you are free to do so. Note! It’s not ok to claim that you’ve developed Cocos2DX or Unity3D :)

- Please remember that we want to be able to test your game without having to install a bunch of obscure windows libs…

Evaluation

- We will look at your source code to evaluate
  - Overall structure and organization of code
  - Performance & memory considerations

- Evaluate how the test performs against the specification

- Evaluate the ”feeling” when playing the game i.e. Attention to transitions, animations and other details that might enhance the user experience

Submitting

- Please send the test back as a .zip file or package to a Google doc link, because sometime our mail server might have problems with some attached docs.

Timeframe

- There's no time limit on it per se, but most people return the solution within 7 days. Please do your best to return the test within 7 days from receiving it, and if you need more time please send us a notification email.