Ascii puzzle game for web browsers
Erkki Luuk†

0.1 Goal
The project involves developing a browser game consisting of a series of puzzles. Each puzzle is a webpage, and solving the puzzle takes you to the next one. There would be 20...50 puzzles/pages in total, progressing from trivial to complex. The puzzles would be as diverse as possible. Formally, they would be based on text, chars and/or ascii art (there may be also a limited use for moving chars, either in-game or by the player). Contentwise, the puzzles would range from finding inconspicuous or hidden links, locating and/or combining diverse clues, solving anagrams, crypto- and steganography tasks, etc. The player can use keyboard and pointing device. At least in some puzzles this would involve text input in address bar. (Although this might be more complex programmatically, input in address bar seems preferable to dedicated text boxes even for UX considerations.) For the same reason – simplicity and unity of style – everything, although otherwise quite diverse, should be ascii-based. Here’s an idea of what some graphical puzzles might look like:

Others might take the form of tabular data, fiction, etc., where the player must locate a clue to progress. Theoretically, based on the prototype, it would be easy to construct games tailored for specific purposes, ranging from IQ tests to “publishing platforms” for weird texts to learning a (programming) language or maths. However, the prototype (the game to be developed) would be concerned w/ generic puzzles of maximum diversity within the bounds given above. I would provide the puzzles. The dev-s are welcome to share their ideas and suggestions but I’d have the final say on what goes into the official prototype.

0.2 Technical
The game should work on all major browser (Chrome, Chromium, Firefox, Safari, IE (Opera?).) As for languages, a HTML/CSS and Javascript combo would probably do the job (but some server-side tinkering may be necessary, and the developers can get to decide on languages themselves). While the progress in the game can be made only by solving puzzles‡, moving backwards through the pages should be doable by inputting <URL>/(1|2|...) in the address bar (up to the page number where the player has progressed to). Some kind of save/restore game system, capable of identifying individual players (including those sharing a computer), is also necessary.

0.3 License
FreeBSD. My name should appear first in the copyright list.

† Specialist of Language Technology, ICS UT (erkkil@gmail.com).
‡ This presents the problem of “solving” puzzles by simply looking at pages’ HTML/JS code. Although the problem would be superficially “solvable” by discriminating between fair play and cheating, a technical solution to the problem should be attempted, e.g. by means of page code being hidden from or obfuscated for the player.