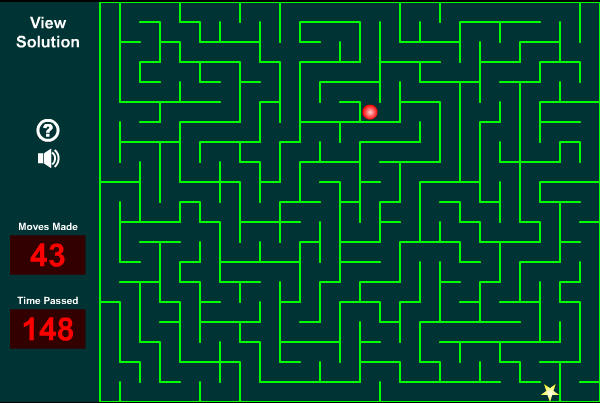
**Projects Analysis**

**“Maze”**

[www.coolmath-games.com/0-maze](http://www.coolmath-games.com/0-maze)

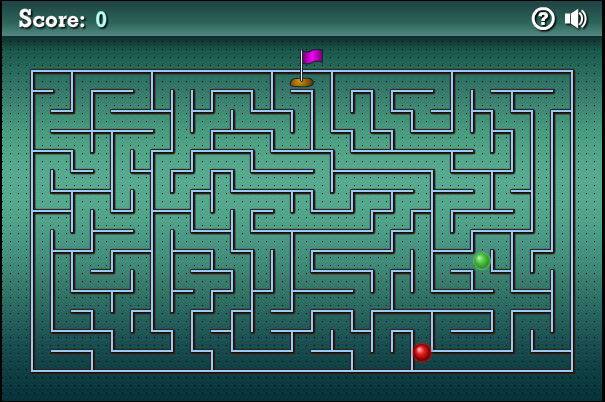


* Works well
  1. While trying several levels no bugs were experienced. When reaching the star, game clearly demonstrates that the game is won and provides a choice of going back to the menu or to play again.
* Visually a good maze design
  1. Chosen maze design is not only visually kind of appealing but also works well on technical level – walls are quite deceiving and makes the game a bit harder. The player and the goal icons stand out well in the maze. Also extra information on the left side of the screen is easily readable.
* Good maze levels
  1. At first maze levels in this game feel quite challenging (though not extremely hard). Also every time played there is a new random maze level. Looks like the mazes might be randomly generated, so the player always has a different experience when playing this game.
* Good additional features
  1. Though the game itself is quite traditional, it also has some extra features not all maze games have. There is a well working timer to see how long it takes for the player to finish the game, plus a steps counter which counts how many moves a player has made. All of this is clearly represented on the left side of the screen. To add to that a “View solution” option is also included – it clearly shows the quickest way for the player to reach the star.
* Clear instructions
  1. At the beginning of the game the controls of the game are clearly explained. The instructions can also be view during the middle of the game (the game pauses).
* Decent sound effects
  1. The game also makes simple sound effect whenever a player moves. Though it is nothing spectacular, it drastically improves the enjoyability of the game.
* Lacks replayability
  1. A player cannot choose harder maze levels which players would want after getting used to the game. To add to that the timer is not very useful when a player cannot play the same levels (always a random maze level) and compare his times.

Overall “Maze” is a decent tradition maze game which has some good extra features, but is not the game you would expect to play for ages – at first a player might be overwhelmed by the complex look of the mazes, but after a while it becomes a bit easier. And there are no harder maze levels which could make this game more replayable.

**“A Maze Race”**

[www.primarygames.com/puzzles/mazes/mazerace](http://www.primarygames.com/puzzles/mazes/mazerace)

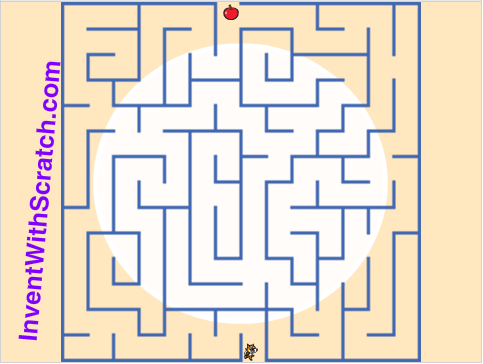


* Great game idea
  1. The game takes the traditional maze game idea and turns it into a race against the AI. The player has to finish the maze before the AI does. The idea works really well, because each time the player wins, AI becomes faster and more challenging. This race idea also adds the feeling of urgency to the game. On some occasions the end result is quite close, which does add a lot of excitement to the game.
* Works well
  1. While trying several levels no bugs were experienced. Both the player and the AI can move without any problems.
* Visually a good maze design
  1. Visually the game looks nice. Has some texture in it. Both the player and the AI icons stand out easily, as well as the goal icon. The design though makes the maze slightly easier to read, but the game is already quite challenging so that is not a problem.
* Good maze levels
  1. Good unique and random maze levels. Each of them are quite challenging and take a bit of time to find the route to the goal.
* Has high value of replayability
  1. Though the game feels a bit easy at the first few levels, the AI becomes really challenging after a while which brings a lot of excitement to the player. Each time playing this game there is a desire to try and win as many rounds as possible before the AI becomes too fast and beats the player.
* Annoying sounds effects
  1. In my opinion, the sounds effect for moving was chosen poorly because they are quite annoying when playing this game.
* Weird movement mechanics
  1. Movement in this game is made to feel very smooth, which is quite nice (not block to block teleporting as in most simple maze games) but it becomes a bit annoying when trying to quickly turn around corners. It is difficult to judge movements in this game while trying to go as quick as possible, which often makes the player get stuck in some corners. This does not really work well in this kind of competitive game.

Overall “Maze Game race” is a real delight. The idea of a race against the AI makes this game extremely enjoyable and exciting. There are only few small problems like annoying sound effects and weird movement mechanics. But these problems do not ruin the overall enjoyability of this game.

**“Maze Game”**

<https://scratch.mit.edu/projects/11710850>



* Decent visual look
  1. Though nothing too amazing, visuals of the game look quite good. Everything stands out quite easily, and the player icon is quite cute (especially how its feet move when walking)
* Works well
  1. While trying the game several times, no bugs were experienced, except for one tiny bug – sometimes when the player icon is too close to the walls it might walk around a bit slower.
* Smooth movement mechanics
  1. The movement is really smooth in this game (not the usual block to block teleporting movement). Easy to walk around corners.
* Random maze levels
  1. This game as well has random new maze level each time you start a new game.
* Maze levels too easy
  1. Maze levels are not challenging at all. They can be completed quite quickly without any challenge.
* No extra features
  1. This game is just a traditional maze game without any other extra features. That really ruins the replayability of this game.

Overall this game is nothing special. Though it visually looks decent and has nice movement mechanics, it is simply too easy and not very enjoyable. Pretty much all mazes can be completed in around 10 seconds and with no extra features there is no desire to play this game after just a few attempts.

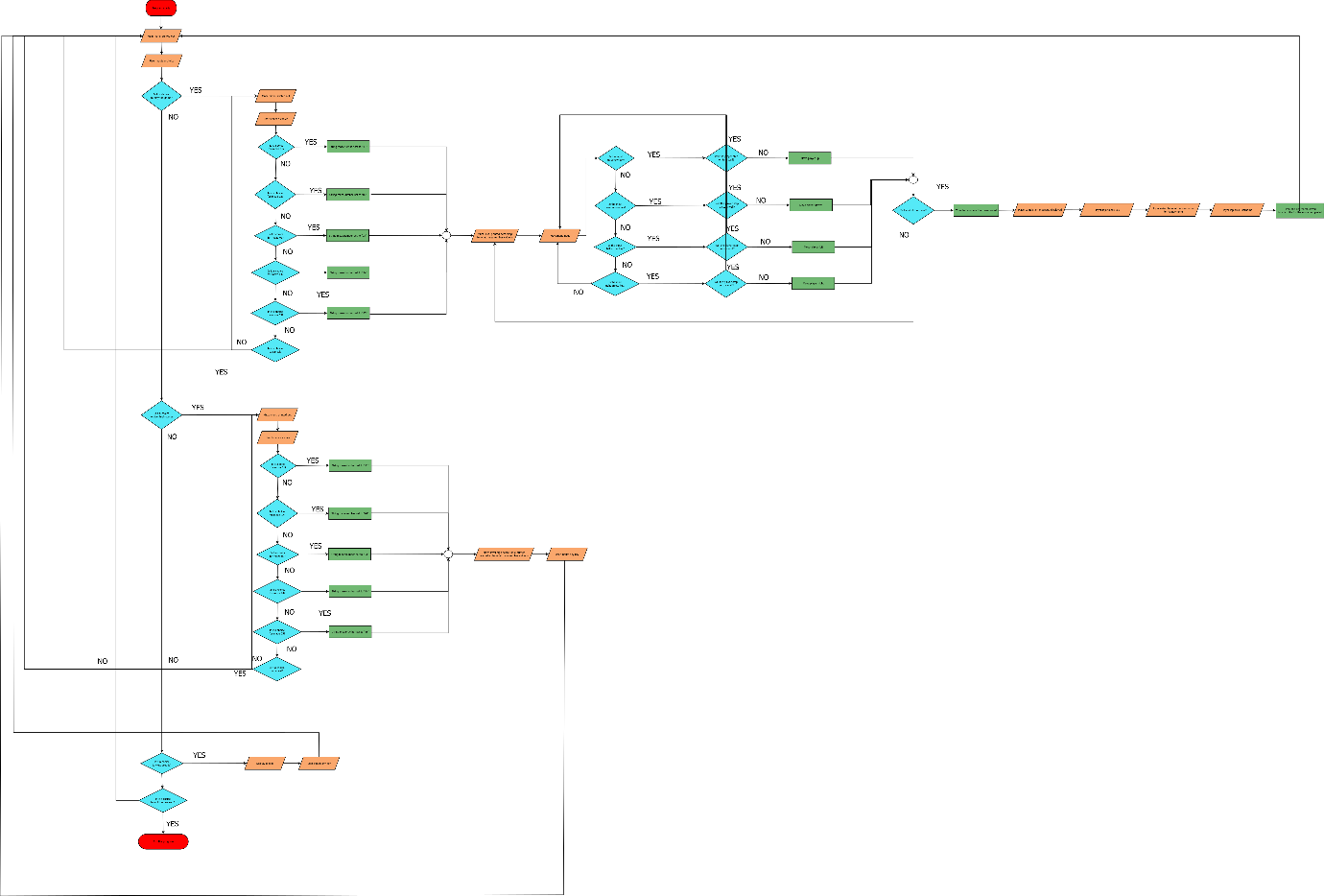
Program Requirements

1. Player can walk 4 directions using keys input.
2. Player cannot cross the walls.
3. Maze must be declared completed when reaching the end of the maze.
4. Program must be able to load several different mazes.
5. Games must be timed.
6. Times must be added to the high scores table.
7. High scores table can be easily accessed by the user.
8. User can input a name when a score is reached.
9. Different mazes must have different high scores tables.

Program Specifications

* Player can walk 4 directions using keys input.
  1. User moves around the maze with using arrow keys. No need to press enter for each input.
* Player cannot cross the walls.
  1. If user tries to move through the wall, his movement input will be ignored.
* Maze must be declared completed when reaching the end of the maze.
  1. When the player reaches the end of the maze, a new output will appear on the console screen, declaring the maze completed.
* Program must be able to load several different mazes.
  1. Player will be able to choose different maze levels through a maze menu.
* Games must be timed.
  1. Timer will start when a maze level is loaded and finish when player completes the maze. Time will be shown to the player straight after the maze is completed.
* User can input a name when a score is reached.
  1. After finishing the maze, the program will ask for the user to input his username. Maze game time will be saved under his username.
* Times must be added to the high scores table.
  1. All times, including all usernames, will be added to a high scores table (probably just a simple text file).
* High scores table can be easily accessed by the user.
  1. User can easily access the high scores table for each level by choosing it through the menu. Results output must be easily clear and readable.
* Different mazes must have different high scores tables.
  1. Each maze level will have a separate high scores table. When a player finishes a maze level, the time will be added to the text file which belongs only to that maze.

**Problem Decomposition**

**Flowchart**