The game is played using:

* a *decoding board*, with a *shield* at one end covering a row of four large holes, and twelve (or ten, or eight, or six) additional rows containing four large holes next to a set of four small holes;
* *code pegs* of six different colors (red, green, blue, black, yellow, white), which will be placed in the large holes on the board; and
* *key pegs*, some colored red and some white, which are smaller than the code pegs; they will be placed in the small holes on the board.

The computer is by default the *codemaker* and the player is the *codebreaker*.  The computer (codemaker) chooses a pattern of four code pegs. Duplicates are allowed, but no blanks. The codemaker may even choose four same-colored code pegs. The computer sequence selection is hidden at the bottom of the board and can be revealed either:

1. upon the codebreaker correctly guessing the combination.
2. upon unsuccessfully reaching the maximum number of tries (ie. how many rows are displayed on the board).
3. or upon pressing the Reveal button.

In all cases, the game is over when the combination is revealed.

The codebreaker tries to guess the pattern, in both order and color, within a maximum of twelve turns. Each guess is made by placing a row of code pegs on the decoding board. Once placed, the codemaker provides feedback by placing from zero to four key pegs in the small holes of the row with the guess. A colored key peg is placed for each code peg from the guess which is correct in both color and position. A white key peg indicates the existence of a correct color code peg placed in the wrong position.[[5]](https://en.wikipedia.org/wiki/Mastermind_(board_game)#cite_note-5)

If there are duplicate colors in the guess, they cannot all be awarded a key peg unless they correspond to the same number of duplicate colors in the hidden code. For example, if the hidden code is red-red-blue-blue and the player guesses red-red-red-blue, the codemaker will award two colored key pegs for the two correct reds, nothing for the third red as there is not a third red in the code, and a colored key peg for the blue. No indication is given of the fact that the code also includes a second blue.[[6]](https://en.wikipedia.org/wiki/Mastermind_(board_game)#cite_note-6)

Once feedback is provided, another guess is made; guesses and feedback continue to alternate until either the codebreaker guesses correctly, or all rows on the decoding board are full.

The game is won upon correctly guessing the combination.[[7]](https://en.wikipedia.org/wiki/Mastermind_(board_game)#cite_note-7)

Pseudo code

1. initialize the board, display buttons, title and other messages
2. when the codebreaker presses play, the codemaker creates the secret combination and a new button appears next to the guess line
3. the codebreaker selects colors for each peg cycling through the colors
4. when all pegs have a color, the codebreaker submits its guess by pressing the corresponding button
5. if the correct sequence has been guessed then the codemaker reveals the bottom, hidden sequence and display the message “you have won” and the game ends.
6. the message stays till the player wants to play again and the game restarts
7. otherwise if the sequence was not correctly guessed, the computer places the corresponding key pegs
8. the players can then submit a new guess or press the Reveal button
9. if a new guess is submitted then the process at 5 starts again
10. or if the reveal button is pressed the game ends, the correct sequence is displayed and the message “you have not guess correctly” is displayed.
11. The message stays till the player choses to play again.

Refined Steps:

The logic starts with establishing the board, the Play button (enabled), the Submit button (disabled) and the Reveal button (disabled).

First render the board with all Peg(s) and KeyPeg(s) greyed out. The board pegs are numbered 0 to 3, for each row, and the keyPegs 0 to 3 for the corresponding row. For future dynamic purposes, I establishing global constants holding maximums. The maximum number of tries for each game is 12 and that corresponds to the equivalent number of playing rows.

The game starts when the Play