

Assignment #2

Deadline: Monday 12/02, 23:59

Mastermind game: We want to develop a traditional game called: *Mastermind*¹. In this game, we have pegs of multiple colors. Player 1 (here, the computer) is the code maker and will choose 4 pegs. For example, it could be (grey, blue, red, purple) (see picture below). Also notice that the order matters and colors can be repetitive. Player 2 (here, the user) is the code breaker and is not aware of the pegs chosen by the code maker. Player 2 has multiple (12 in the picture below) chances to get to the Player 1's choice. Player 2 starts to guess the 4 pegs and for each iteration of guesses, Player 1 gives points to Player 2 according to how close the guess was to the Player 1's choice. The points are white and red pins. For each correct color but wrong order, Player 2 gets a white pin and if the order of that color is also correct, Player 2 receives a red pin, and no pin otherwise.



Source: Wikipedia

We would like to implement this game with a minor modification: Instead of colors, we assume that the pegs can take different integer numbers between 0 to 9 (inclusive). Player 2 also decides about the maximum iterations allowed in each round. Below, is a more detailed description of how to implement the game:

NOTE1: The format of your output must be identical to the example included in the last page and as also explained in the description. Points will be deducted, otherwise.

NOTE2: You MUST use arrays for the computer choice and the user input choice.

NOTE3: You MUST follow the 'good programming style'.

NOTE4: A hint: use one scanner variable per TYPE (and not per variable) if your program does not work properly with one scanner for all user inputs.

NOTE5: Please name your files as A2_<st#>.java. For example, for a person with student# 000000 the file name would be A2_000000.java. Submit your files on your dropbox folder on Canvas.

¹ [https://en.wikipedia.org/wiki/Mastermind_\(board_game\)](https://en.wikipedia.org/wiki/Mastermind_(board_game))

I. The game starts with a welcome message identical to the following

```
*****
*****
*****Mastermind*****
*****
*****
```

Manual:

- (1) Computer chooses a 4-tuple integer values between 0-9
 - (2) Order matters in the 4-tuple and can be repetitive
 - (3) You can set the number of iterations (guesses) per game
 - (4) In each iteration you guess the 4-tuple
 - (5) Enter four numbers with space in between and enter when done
- ```

```

- II. The computer randomly picks 4 integer numbers (A[0], A[1], A[2], A[3]), each of which between 0 to 9 and could be repetitive (choose better names though).
- III. The computer asks about the number of iterations Player 2 (user) wants to have in the game and takes an integer from the user as the maximum number of iterations. The user input must be in the same line as the question (see the example below).
- IV. The computer asks the user to guess the 4-tuples. The user input must be in the same line as the question, a space between each two numbers, and an enter after the last one (as shown in the example).
- V. The computer reports the number of white and red points (as shown in the example).
- VI. Each round of the game finishes if one of the following happens:
  - a. If the user cannot guess the 4-tuple code before the maximum number of iteration of guesses reaches, the computer returns the following message: 'I am sorry. You failed.' Then, the computer's choice is printed in the console. (see the example for formatting).
  - b. If the user can guess the code, the computer returns the following message: 'Congratulations! You could decode it.'
- VII. The computer asks if the user wants to continue with another round
  - a. If the answer is 'N', the game terminates with a game over message (as shown in the example).
  - b. If the answer is 'Y', the computer goes back to Step (II).

Here is an example of an easy case of grading a user input with red and white pins:

| Computer choice | User choice  | red | white |
|-----------------|--------------|-----|-------|
| (1, 4, 3, 8)    | (2, 4, 8, 5) | 1   | 1     |

Here are some examples of more challenging cases:

| Computer choice | User choice  | red | white |
|-----------------|--------------|-----|-------|
| (2, 6, 6, 7)    | (5, 6, 7, 8) | 1   | 1     |
| (2, 6, 6, 7)    | (6, 6, 6, 1) | 2   | 0     |
| (1, 2, 3, 4)    | (1, 1, 1, 1) | 1   | 0     |

Example Please note that the text in red is the user input

```


***** Mastermind *****

Manual:
(1) Computer chooses a 4-tuple integer values between 0-9
(2) Order matters in the 4-tuple and can be repetitive
(3) You can set the number of iterations (guesses) per game
(4) In each iteration you guess the 4-tuple
(5) Enter four numbers with space in between and enter when done

How many iterations do you want to have? 3
Iteration 1. Next guess: 1 3 5 8

| Your choice || white | red |

| (1, 3, 5, 8) || 1 | 0 |

Iteration 2. Next guess: 0 9 6 1

| Your choice || white | red |

| (0, 9, 6, 1) || 0 | 1 |

Iteration 3. Next guess: 0 7 6 3

| Your choice || white | red |

| (0, 7, 6, 3) || 0 | 1 |

I am sorry. You failed.
Player 1 choice was :

(0, 8, 4, 4)
Do you want to play another round? (Y/N) Y
How many iterations do you want to have? 1
Iteration 1. Next guess: 7 6 5 4

| Your choice || white | red |

| (7, 6, 5, 4) || 1 | 0 |

I am sorry. You failed.
Player 1 choice was :

(4, 4, 1, 3)
Do you want to play another round? (Y/N) N

***** Game Over! *****


```