

Text-based Snake game

Implement the classic [Snake](#) game in Kotlin or Java using the terminal (command line) for the graphic representation. You are allowed to use third-party libraries for printing on the terminal such as [Lanterna](#), but the code design and implementation should be entirely yours, copying parts or complete solutions from the Internet is strongly discouraged.

Rules of the game

The game field is represented by a two-dimensional array, each element in the array is one of the game objects:

- Brick - #
- Grass - " " (empty space)
- Snake segment - *
- Apple - o

A snake is a sequence of snake segments. At the beginning of the game, the snake consists of 3 segments. It grows by eating apples with its head. Each apple adds a new segment at the tail of the snake when the snake's head moves over the apple's position. The apples appear at a random unoccupied (grass) position on the field and stay for 20 moves, then disappear and a new apple appears. Only one apple can exist in a single moment. A horizontal or a vertical sequence of bricks form a wall. The game field is surrounded by walls, you can place additional walls inside the field by your design. The game ends when the snake hits a wall or its body.

For the sake of simplicity, there is only one snake in the field.

Bonus task

Add a special game object Pear - d. Pears are similar to apples, but unlike apples, pears reverse the movement direction of the snake. Apples and pears appear on a random basis with a probability of 5 to 1 (apples to pears).

Input

The snake is controlled by the arrow buttons of the keyboard `← → ↑ ↓`.

The game field should be loaded from a file, which contains the field's width on the first line, the field's height on the second line, followed by the bricks' positions. We assume that the terminal uses a monospaced font, i.e. all symbols have the same width in pixels. The relative path to the field's file should be the first and only parameter accepted by the program.

Example:

> Enter field file path: ./Field1.txt

```
----- Field1.txt -----
30
10
#####
#                                     #
#      #####                       #
#          #                       # #
#          #                       # #
#          #                       # #
#                                     #
#                                     #
#                                     #
#                                     #
#####
-----
```

Source code requirements

You should keep your source code well formatted and easy to understand, meaningful comments will be appreciated. It is recommended to use a version control system, preferably Git.

Send your source code as a .zip file and/or public repository URL. Add a README.md file in the root directory which contains instructions for running the program and any information you find appropriate.

Deadline: 7 days, starting from the day the task is received.