WEEK 1

ONDERWERPEN

- multi-threading
- Java FX
- Lambda expressies

OPMERKING

Alle opgave nummers verwijzen naar Liang, **10e editie**. Om verwarring te voorkomen (verschillende uitgaven van het boek) zijn de opgaven uit Liang hier voor een deel overgenomen, maar dit is **niet** altijd compleet. Voor de volledige versie moet je het boek van Liang raadplegen.

OPGAVEN MULTI-THREADING

Hoofdstuk 30 pag. 1134 Programmeer opgaven.

30.1 Rewrite Listing 30.1 to display the output in a text area, as shown in Figure 30.30.
Create a TextArea and use appendText() to add text.

30.4 (*Synchronize threads*) Write a program that launches 1,000 threads. Each thread adds 1 to a variable sum that initially is 0. Define an Integer wrapper object to hold sum. Run the program with and without synchronization to see its effect.
You can use newFixedThreadPool() to create a fixed number of threads in the pool.

30.6 (*Bouncing balls*) Rewrite and combine listings 15.17 BallPane.java and 15.18 BounceBallControl.java. (You can find these files on Blackboard). Make a separate thread to animate bouncing ball movements. First make a JavaFX application using BallPane.java, where you register handlers for mouse and key events. In the class BallPane create a thread and use `Platform.runLater(() -> moveBall())` and add a delay with Thread.sleep() that can be controlled from the from the main GUI-thread (controller). In Listing BallPane.java you have to modify methods pause/play and increaseSpeed/decreaseSpeed.

30.8 (*Account synchronization*) Rewrite Listing 30.6, ThreadCooperation.java (see Blackboard), using the Account-object's **wait()** and **notifyAll()** methods. Note: an object becomes a monitor once a thread locks it.

30.15 (*Parallel sum*) Implement the following method using Fork/Join to find the sum of a list:

```
public static double parallelSum(double[] list)
```

Write a test program that finds the sum in a list of 1000 double values.

Note : this is quite similar to LISTING 30.11 ParallelMax.java (see Blackboard). But now you create a private static class SumTask extends RecursiveTask<Double> { }.

OPGAVEN JAVA FX

Hoofdstuk 14 pag. 578 Programmeer opgaven.

14.2 (*Tic-tac-toe board*) Write a program that displays a tic-tac-toe board, as shown in Figure 14.43b. A cell may be X, O, or empty. What to display at each cell is randomly decided. The X and O are images in the files x.gif and o.gif. You should use a GridPane, and you can use Math.Random() to place either X or O.

14.6 (*Game: display a checkerboard*) Write a program that displays a checkerboard in which each white and black cell is a Rectangle with a fill color black or white, as shown in Figure 14.44c.

14.16 (*Display a* 3 * 3 *grid*) Write a program that displays a 3 * 3 grid, as shown in Figure 14.47c. Use red color for vertical lines and blue for horizontals. The lines are automatically resized when the window is resized. Use pane.widthProperty() and pane.heightProperty() to set endpoints of the lines.
Resizing is accomplished by binding properties, e.g. "line1.endXProperty().bind(pane.widthProperty());"

Hoofdstuk 15 pag. 589 Review vragen.

15.2 Can a button fire a MouseEvent? Can a button fire a KeyEvent? Can a button fire an ActionEvent?

15.3 Why must a handler be an instance of an appropriate handler interface?

15.4 Explain how to register a handler object and how to implement a handler interface.

15.6 What is the registration method for a button to register an ActionEvent handler?

15.13 Show the output of the following code: (page 599).

15.18 How do you set focus on a node so it can listen for key events?

15.24 How does the program make the ball moving?

15.25 How does the code in Listing 15.17 BallPane.java change the direction of the ball movement?

15.26 What does the program do when the mouse is pressed on the ball pane? What does the program do when the mouse is released on the ball pane?

15.27 If line 32 in Listing 15.18 BounceBallControl.java is not in the program, what would happen when you press the UP or the DOWN arrow key?

15.28 If line 23 is not in Listing 15.17, what would happen?

Hoofdstuk 15 pag. 621 Programmeer opgaven.

15.4 (*Create a simple calculator*) Write a program (with JavaFX) to perform addition, subtraction, multiplication, and division, as shown in Figure 15.25a.
(Zie ook het voorbeeld op Blackboard gemaakt met Swing).

Hoofdstuk 16 pag. 662 Review vragen.

16.37 When the game starts, what value is in whoseTurn? When the game is over, what value is in whoseTurn?

16.38 What happens when the user clicks on an empty cell if the game is not over? What happens when the user clicks on an empty cell if the game is over?

16.39 How does the program check whether a player wins? How does the program check whether all cells are filled?

Review vragen over LISTING 16.13 TicTacToe.java regel 28 pag. 659:

- what does line 28 "pane.add()" exactly do?
- why are j and i reversed in order in pane.add(cell, j, i)
- what is the type of cell?
- what is the type of Cell?

Review vraag over LISTING 16.13 TicTacToe.java pag. 658:

Another design could be to have 3 classes: Model, View and Controller. Would this be a better design than the solution offered by Liang with two classes TicTacToe and (inner class) Cell? Motivate your answer. Which design criteria do you think are applicable here?

OPGAVE LAMBDA EXPRESSIES

Programmeer opgave.

Op Blackboard vind je de files Testsorting.java en Developer.java. De bedoeling is dat je de lijst developers sorteert op leeftijd met behulp van een Lambda expressie.