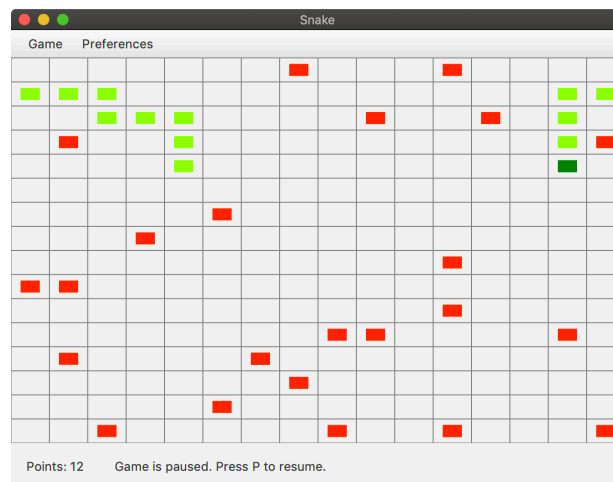


GUI mit JavaFX

Aufgabe 1 Snake

24 Punkte

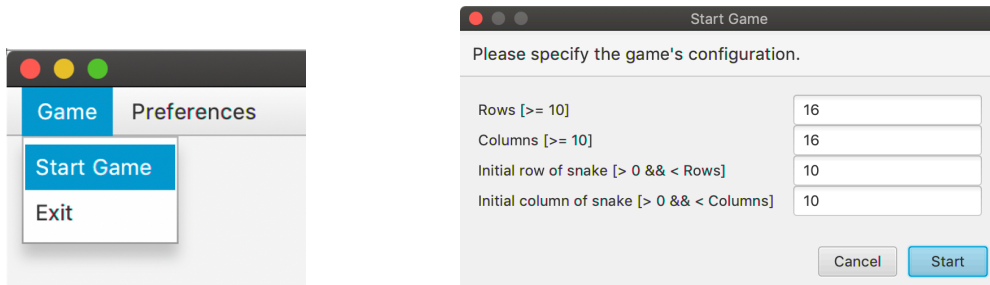
Implementieren Sie das klassische Spiel „Snake“ als JavaFX Anwendung. Snake wurde auf zahlreichen Mobilgeräten vor der Smartphone-Ära mitgeliefert und definiert ein einfaches Spielkonzept: Es wird eine Schlange auf einem zweidimensionalen Spielfeld gesteuert, die auf selbigem Spielfeld verteilte Nahrung verspeisen kann. Die Schlange bewegt sich dabei in einem konstanten zeitlichen Intervall in eine Richtung und wird mit jeder verschlungenen Nahrung mit Punkten und einem verlängerten Schlangenkörper belohnt. Ziel des Spiels ist es, möglichst viele Punkte zu sammeln, ohne mit dem Schlangenkopf den eigenen Körper zu berühren. Mit jedem gesammelten Punkt wächst die Schlangenlänge und belegt somit mehr Platz am Spielfeld, was ein zunehmend taktischeres Vorgehen bei Ausweichmanövern erfordert.



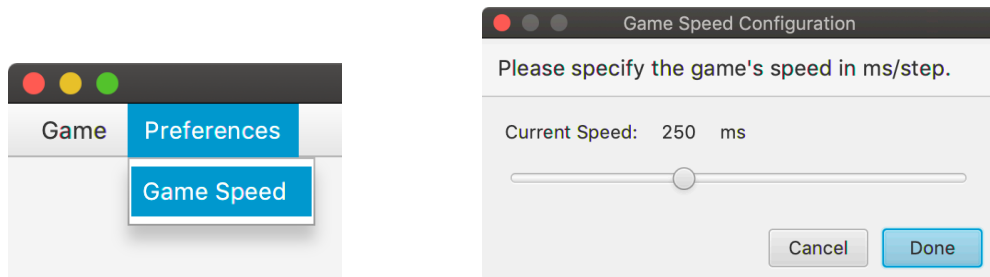
Die im Screenshot gezeigte Applikation ist eine beispielhafte Darstellung eines möglichen Interfaces. Am oberen Ende des Fensters befindet sich eine Menüleiste, über die das Spiel gestartet, beendet und konfiguriert werden kann. Am unteren Ende zeigt ein Balken die aktuelle Punktzahl sowie den Spielstatus an. Im zentralen Bereich der Applikation befindet sich das zweidimensionale Spielfeld mit leeren Feldern, roten Feldern (Nahrung) und grünen Feldern (Schlange, wobei dunkelgrün den Kopf und hellgrün den Körper darstellen). Die Eingabemöglichkeiten können folgend zusammengefasst werden:

- Die Schlange bewegt sich automatisch in einem Intervall von x Millisekunden in eine definierte Richtung, wobei x konfigurierbar ist
 - Würde der nächste Spielzug außerhalb des Spielfelds liegen, erscheint der Kopf der Schlange am gegenüberlegenden Ende des Spielfelds
 - Beispiel: Die Schlange bewegt sich wie im Screenshot oben links aus dem Spielfeld hinaus und erscheint am rechten Ende erneut
- Die Richtung kann vom Spieler per Drücken der Pfeiltasten geändert werden
- Die Schlange kann sich nur horizontal und vertikal bewegen, diagonale Spielzüge sind nicht möglich
- Das Tippen der Taste ‚P‘ pausiert das Spiel und lässt es nach einem weiteren Tippen wieder weiterlaufen

Beim Start des Spiels über das Menü soll ein Dialog erscheinen, der die Spielfläche und den Startpunkt der Schlange spezifiziert.



Achten Sie dabei darauf, dass die einzelnen Textfelder auf Sinnhaftigkeit überprüft werden! Der Startpunkt der Schlange darf nicht außerhalb der Grenzen des Spielfelds liegen und Eingaben von invaliden Zahlen bzw. Buchstaben soll verhindert werden. Für diesen Zweck könnte man beispielsweise den „Start“ Button deaktivieren, sofern invalide Daten in den Textfeldern stehen. Die Spielgeschwindigkeit soll zusätzlich über einen Dialog eingestellt werden können.



Das Spiel soll laufend neue Nahrungsfelder erscheinen lassen, die konkrete Strategie hinter der Positionierung und der Frequenz bleibt jedoch Ihnen überlassen. Eine mögliche Implementation wäre beispielsweise das Hinzufügen eines zufällig positionierten Nahrungsfelds wenn die Schlange eine bestimmte Anzahl von Feldern durchquert hat, ohne dass ein Nahrungsfeld angetroffen wurde. Kommt das Spiel zum Ende, soll ein Dialog mit den erreichten Punkten und einem kurzen Abschlusstext erscheinen. Technisch sollen weiters folgende JavaFX Features während der Implementation der Applikation verwendet werden:

- FXML als Technologie für mindestens eine Komponente (als Alternative zu einer rein codebasierten Implementation der GUI)
- Styling mit Hilfe von CSS für mindestens eine Komponente
- Property Binding überall, wo es sinnvoll erscheint

Für diese Aufgabe können Sie Zusatzpunkte ergattern, die der Gesamtpunktzahl Ihrer gesammelten OPR Übungen angerechnet werden können.

Zusatzaufgaben (je +1,5Punkte)

Eine Auswahl von möglichen Zusatzfeatures wäre:

- Unterschiedliche Typen von Nahrungsfeldern, die eine variable Anzahl an Punkten geben
- Ansprechenderes Design des Spielfelds, der Schlange und der Nahrungsfelder
- „Power-Up“ Felder, die kurzzeitig die Spielgeschwindigkeit ändern
- ...

Eventuell suchen Sie sich Features aus bekannten Snake Implementationen, die Sie umsetzen können. Die Anzahl der Bonuspunkte beträgt 1,5 Punkte pro Zusatzfeature, die endgültige Punkteanzahl wird jedoch letztendlich von den Tutoren festgelegt. Denken Sie daran, dass Ihre implementierten Zusatzfunktionen auch für die Tutoren sichtbar sein müssen!

Hinweise

JavaFX kennt das Konzept von „Layouts“, die es Entwickler*Innen erlauben die verwendeten GUI Komponenten automatisch nach einem vordefinierten System anzuordnen. Eine ausgezeichnete Zusammenfassung (inkl. kurzer beispielhafter Verwendung) der klassischen Layouts ist unter https://docs.oracle.com/javafx/2/layout/builtin_layouts.htm zu finden. Die Klassen **BorderPane**, **GridPane**, **HBox**, **VBox** und **AnchorPane** werden vermutlich am nützlichsten sein.

Um Nutzereingaben auf der Tastatur abzufangen (z.B.: die Pfeiltasten zur Steuerung der Schlange), können Sie die **onKeyPressed** Methode von JavaFX Komponenten verwenden. Dort kann eine Funktion übergeben werden, die vom JavaFX Framework bei Tastenanschlägen für Sie aufgerufen wird. In dieser sogenannten „Handler“ Funktion können Sie dann situationsbedingt auf die Eingaben reagieren. Falls die Tastatureingaben nicht korrekt erkannt werden, müssen Sie unter Umständen noch die **requestFocus()** Methode der Spielfeldkomponente aufrufen, damit JavaFX die Eingaben an diese Komponente weiterleitet.

Das Spielfeld kann auf verschiedenste Art und Weisen realisiert werden. Einerseits wäre eine Implementation als GUI Komponente vorstellbar (z.B.: **GridPane** mit kleineren Panes oder **ImageViews** als Spielfelder), andererseits kann das Spielfeld jedoch auch rein mit der **Canvas** API von JavaFX gezeichnet werden! Sehen Sie sich gegebenenfalls die Dokumentation unter <https://docs.oracle.com/javase/8/javafx/graphics-tutorial/canvas.htm#JFXGR214> dazu an.

Um den automatischen Ablauf der Spielzeit (bzw. der Spielschritte) zu implementieren, können Sie entweder die **Timeline** oder die **AnimationTimer** Klasse verwenden. Beide Optionen können in einem vordefinierten zeitlichen Intervall eine zuvor angegebene Funktion für Sie aufrufen, in der Sie die Möglichkeit bekommen, den Spielstatus zu aktualisieren.

Ein gutes Tutorial, wie JavaFX in Eclipse zu verwenden ist, ist hier zu finden:

<https://openjfx.io/openjfx-docs/#introduction>