

Milestone 2 (ABGABE: 06.12.2017)

Das Spiel Snake mit einem Zellulären Automat

[https://de.wikipedia.org/wiki/Snake_\(Computerspiel\)](https://de.wikipedia.org/wiki/Snake_(Computerspiel))

Regel: Zellen verändern ihren Zustand unter Berücksichtigung anderer Zellen
(Nicht wie bei GameOfLife durch eigene Bedingungen)

Aufgabe 1. Bewegung von Zellen (25 Punkte).

a) (15 Punkte)

Schreibe eine (Evolutions-) Funktion, wobei man lebende Zellen horizontal und vertikal bewegen kann. Verwende ein torusförmiges Feld, wie in Game of Life. Die Richtungen der Bewegungen hängen von einer bestimmten Nummer ab, welche in einem CA-Array gespeichert sind.

(Beispiel: 2 – unten, 8 - oben, 4 - links, 6 – rechts)

b) (10 Punkte)

Füge Grenzen/Ränder in das Feld hinzu. Falls ein Objekt einen Rand erreicht, so bewegt sich diese Figur **nicht** weiter (Kollisionen zwischen Objekte sind hier nicht enthalten. Es geht nur um Rand vs. Objekt).

Aufgabe 2. Tastatureingaben und QT(30 Punkte).

a) (5 Punkte)

Füge eine neue oder modifiziere eine vorherige Funktion, um mithilfe der eingegebenen Nummer aus Aufgabe 1a die Richtung eines Objekts zu ändern. (Die Richtungen sollten als Bedingungen gesetzt werden und sollen **nicht** für “body” und “tail” genutzt werden)

Hinweis: Der “Head” vom Snake kennt die Richtung. “Tail” und “Body” bewegen sich dem Head hinterher (siehe Abbildung 1).

b) (5 Punkte)

Füge eine Funktion ***PrepareFieldSnake()*** hinzu, welches das Feld leert und ein Objekt erzeugt, welches sich bewegen lässt (Später der Kopf von Snake)

c) (10 Punkte)

In gamewidget.h/cpp soll es eine Möglichkeit auf Tastatureingaben geben. Implementiere eine Funktion

“void GameWidget::keyPressEvent(QKeyEvent *event)”, welches die Richtung des Objekts per Tastatureingabe ändert. (Richtungsänderung mithilfe der Funktion aus Aufgabe 2a + CA world evolution + aktualisieren des gezeichneten Feldes)

d) (10 Punkte)

Passe die CheckBox an bzw. füge einen neuen Eintrag in die Universe Mode Liste ein, um das Spiel Snake zu starten (Siehe Abbildung 2). Falls Snake gewählt wurde, so soll das Feld vom Zellulären Automaten geleert werden und die Funktion ***PrepareFieldSnake()*** ausgeführt werden.

Aufgabe 3. Snake head, body and tail (25 Punkte).

a) (5 Punkte)

Füge eine Zelle für Tail hinzu, welches sich hinter body von Snake bewegt (Abbildung 1).

b) (5 Punkte)

Aktualisiere die Richtungsänderungsfunktion von Head, um die Bewegung Richtung Tail zu verhindern.

Hinweis: In Abbildung 1 soll der “Head” sich nicht nach links bewegen können, sondern nur nach oben, unten und nach rechts.

c) (10 Punkte)

Füge den “body” mit einigen Zellen dem SnakeObjekt hinzu. Das erste Element von Snake ist der “head”. Das letzte Element ist “tail”. Die Elemente zwischen “tail” und “head” bilden den “body”. Jedes Element erhält in jedem Schritt die Position des vor ihm stehenden Elements

(Beispiel: Das erste Element von Body bekommt als nächste Position die alte Position von Head).

d) (5 Punkte)

Modifiziere die Bewegungsfunktion für Head, um den Zustand “tod” darzustellen. Snake ist genau dann tod, wenn Snake über sich selbst kreuzt oder gegen den Rand stößt (Entlang des Randes kann Snake sich bewegen)

Aufgabe 4. Nahrung (20 Punkte).

a) (5 Punkte)

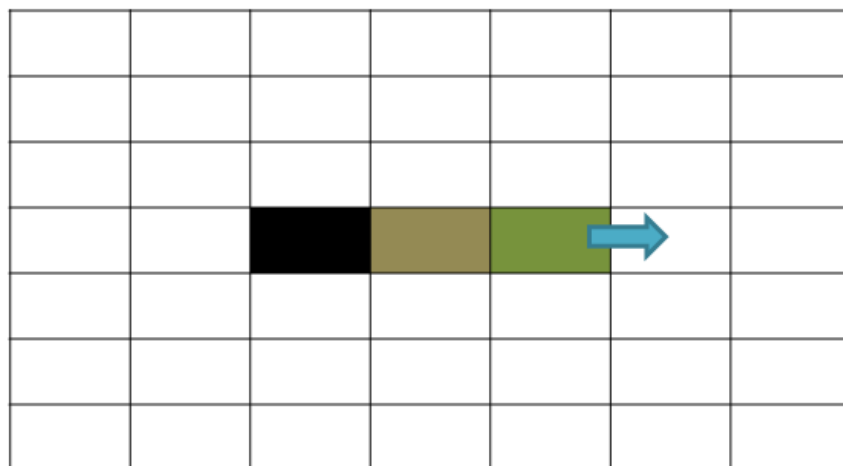
Implementiere eine Funktion, welches “Nahrung” für Snake zufällig in eine leere Zelle setzt.

b) (5 Punkte)

Modifiziere ***PrepareFieldSnake()*** so, dass bei Spielstart Snake und eine Nahrung auf dem Feld sind (siehe Abbildung 2).

c) (10 Punkte)

Modifiziere die Bewegungsfunktion für Head, damit Snake die Möglichkeit hat, Nahrung aufzunehmen. Durch Nahrungsaufnahme nimmt die Größe von Snake zu.



HEAD

BODY

TAIL

HEAD gibt die
Richtung vor

