

Übungsblatt 4

Abgabe bis Dienstag, den 24. Mai 2022 um 12:00 Uhr

Bei diesem Übungsblatt geht es darum, eine einfache Version von John Horton Conways Game of Life zu implementieren. Dabei wird ein zweidimensionales Feld von Zellen (jede für sich „tot“ oder „lebendig“) auf dem Bildschirm angezeigt. Die Werte der Zellen ändern sich nach einfachen Regeln (die in der Vorlesung erklärt wurden) über die Zeit.

Aufgabe 1 (15 Punkte)

Schreiben Sie Dateien `GameOfLife.h` und `GameOfLife.cpp` mit den folgenden Funktionen. Wie schon beim letzten Übungsblatt wurde für einige dieser Funktionen bereits ganz oder teilweise Code in der Vorlesung geschrieben und über das SVN bereitgestellt.

Benutzen Sie für die folgenden Werte globale Variablen: die Dimensionen des Spielfelds, zwei(!) zweidimensionale Felder (eins für den aktuellen Zustand, eins für den jeweils nächsten Zustand), ein Zeiger auf das eine Feld, ein Zeiger auf das andere Feld, die Koordinaten der zuletzt mit der Maus angeklickten Zelle, die Anzahl der schon absolvierten Schritte, die Anzahl der lebenden Zellen im aktuellen Zustand und ob das Spiel gerade läuft oder angehalten ist. Verwenden Sie für die maximale Größe des Spielfeldes in jeder Dimension eine Konstante `MAX_SIZE`. Achten Sie darauf, dass Sie an keiner Stelle in Ihrem Code auf einen Index < 0 oder $\geq \text{MAX_SIZE}$ zugreifen.

`void initTerminal()`: Initialisierung des Terminals und der Spielfelddimensionen mittels `ncurses`.

`void initGame()`: Setzen des Anfangszustands des Spiels und aller zugehöriger globaler Variablen. Zu Beginn sollte keine Zelle lebendig und das Spiel im Zustand „angehalten“ sein.

`void showState()`: Anzeige des aktuellen Zustands auf dem Bildschirm (mit möglichst „quadratischen“ Zellen).

`int numAliveNeighbors(int row, int col)`: Berechnen der Anzahl der lebendigen Nachbarn der gegebenen Zelle.

`void updateState()`: Berechnen des neuen Zustands nach den in der Vorlesung erklärten Regeln des Game of Life. Benutzen Sie dabei die oben erwähnten Zeiger: Wenn der aktuelle Zustand in einem der beiden Felder steht, soll der neue Zustand in dem anderen Feld berechnet werden. Danach tauschen die beiden Zeiger ihre Rollen.

[darüber meditieren, ob Sie eine Figur im Game of Life sind, dann umblättern]

`bool processUserInput(int keycode)`: Verarbeitung des gegebenen Keycodes. Wenn die Leertaste gedrückt wird, soll das Spiel automatisch weiterlaufen bzw. ein automatischer Lauf angehalten werden. Wenn die Taste `s` gedrückt wird (für `step`), soll genau ein Schritt ausgeführt werden. Mit der Taste `q` (für `quit`) soll das Spiel beendet werden. Bei einem Mausklick soll der Zustand der Zelle, auf die der Mauszeiger zeigt, invertiert werden. Wenn die Taste `g` gedrückt wird, soll ein sogenannter „Glider“ an der Stelle des letzten Mausklicks eingefügt werden (der Code dafür wurde bereits in der Vorlesung geschrieben).

Schreiben Sie wie gehabt für jede der Funktionen `initGame`, `numAliveNeighbors`, `updateState` und `processUserInput`.

Aufgabe 2 (5 Punkte)

Schreiben Sie eine Datei `GameOfLifeMain.cpp` mit einer `main` Funktion, die das Spiel auf die beschriebene Weise realisiert. Die Zeit zwischen zwei Schritten beim automatischen Lauf sollte ca. 50 Millisekunden betragen. Der Code für die `main` Funktion sollte wie gehabt kurz sein und auf geeignete Weise Funktionen aus Aufgabe 1 aufrufen. Passen Sie Ihr Makefile vom letzten Übungsblatt (oder von der Musterlösung) so an, dass wie gehabt bei Änderungen immer nur so viel kompiliert wird wie nötig.

Aufgabe 3 (0 Punkte)

Das Spiel lässt sich mit relativ wenig Aufwand auf interessante Arten erweitern. So kann zum Beispiel im angehaltenen Zustand mit der Taste `r` das Spielfeld mit zufälligen Werten initialisiert werden. Oder es können weitere Figuren definiert werden, die durch bestimmte Tasten hinzugefügt werden können. Im WWW findet sich dazu ein reicher Schatz und im Vorlesungscode die Positionen für eine sogenannte „Gliderfabrik“.

Laden Sie wie gehabt alle Code-Dateien und das Makefile in unser SVN hoch, in einem neuen Unterverzeichnis `blatt-04`. Es gelten weiterhin die 10 Gebote und nehmen Sie sich bitte die Seite Fragen auf dem Forum auf unserem Wiki zu Herzen.

Laden Sie wie gehabt auch eine Datei `erfahrungen.txt` hoch (im Unterordner `blatt-04`), in der sie kurz Ihre Erfahrungen mit dem Ü4 und der Vorlesung dazu beschreiben.

Wie waren Ihre Erfahrung mit der „Einführung in die Programmierung“? Insbesondere: Haben Sie die Übungen dort bearbeitet und haben Sie das Gefühl, dort gelernt zu haben, wie man im Prinzip programmiert? Falls Sie die Veranstaltung nicht gehört haben: welche Programmiererfahrung haben Sie?