

Imperative Programmierung

Aufgabenblatt IX

1. Um die reflexive transitive Hülle zu berechnen, kann eine Variation des Floyd-Warshall Algorithmus verwendet werden. Mein Programm liest eine binäre Matrix mit den Abmessungen n und m ein und gibt eine Matrix mit den gleichen Abmessungen zurück.
Das gesamte Programm wurde in „huelle.c“ implementiert.
2.
 - a) Um die bedingten Wahrscheinlichkeiten zu berechnen, muss das Programm sich immer die letzten beiden Buchstaben merken. Dann kann einfach bei einem Buchstaben die Anzahl dieses Buchstaben nach den beiden vorherigen Buchstaben, um 1 erhöht werden. Dazu wird ein 3-dimensionales Array genutzt. Um zu speichern wie viele Buchstaben nach zwei bestimmten Buchstaben im Text vorlagen, wird ein 2-dimensionales Array genutzt. Mit diesen zwei Arrays können dann die prozentualen Anteile berechnet werden.
Das Berechnen der Wahrscheinlichkeiten wurde in „randText.c“ in der Funktion „calculateFreq“ implementiert.
 - b) Um dann einen zufälligen Text zu berechnen, kann einfach das Wahrscheinlichkeits-Array genutzt werden. Dann wird jeder Buchstabe einzeln auf Grundlage der beiden Vorgänger generiert. Dabei wird eine zufällige Fließkommazahl zwischen 0 und 100 generiert und dann auf Grundlage der Wahrscheinlichkeiten ein Buchstabe ausgegeben. Dazu wird jeder Buchstabe mit einer for-Schleife durchlaufen und immer die Wahrscheinlichkeit zu einer Summe addiert. Falls dann die Summe größer als die zufällige Zahl ist, endet die Schleife und der aktuelle Buchstabe wird ausgegeben.
Das Generieren des Textes wurde in „randText.c“ in der Funktion „makeText“ implementiert.