

Übung endrekursive Funktion

- Aufgabe zum endrekursiven Verbinden zweier sortierter Listen zu einer *verbinde-endrekursiv_Erarbeitung.scm* ?

wiederholen und nachholen

- Unterschied in Laufzeit bei Erfolgsfall und Misserfolgsfall → warum?
→ dies ist eine **optimierte Tiefensuche!**
- Satisfizierungsproblem / Optimierungsproblem: *fueller-ts-Optimierung.scm*

Breitensuche

- nur kurz mit Hinweis Datenstruktur Warteschlange statt Stack, Umsetzung aber nicht mit Warteschlange, sondern mit Einsatz von Akkumulator →
- Präsentation **P02a BS Entwicklung und Kohäsion.pdf**

Wegeproblem kürzester Weg

- Präsentation **P03-a Graphen Modell und Datenstruktur.pdf**
- Präsentation **Gallenbacher_Kap1_Ausschnitt.pdf** , Hinweis auf Buch und CD
 - komplett einmal "durchspielen" wäre gut → auf Papier!
 - Eingehen auf Gallenbacher-Konzept ohne Programmierung
 - Hinweis auf : Selbst wenn Gallenbacher Informatik ohne Computer macht, löst er sich dennoch nicht von einer prozeduralen bzw objektorientierten Algorithik !
- Schritt zur Programmierung bedeutet bei komplexen Problemen Modularisierung:
 - Datenstruktur entwickeln
 - Entscheidung Knotenliste oder Kantenliste
für unsere Entwicklung: Kantenliste
- Präsentation **P03-c2 Dijkstra Funktionale Analyse_Kanten.pdf**
- Kantenliste in *gallenbacher-graph-kanten.scm* mit Zugriffsfunktion s.u.
Denn, die Zugriffsfunktionen für die Datenstruktur laut Gallenbacher "*Gib alle Nachfolgestädte der aktuellen Stadt*" wird bei Kantenlisten nicht gebraucht, statt dessen **nachfolge-kante** in *gallenbacher-graph-kanten.scm*
- Erarbeitung von Funktionen gemeinsam im Kurs:
prioritaetswarteschlange_Erarbeitung.scm → *prioritaetswarteschlange_Kurs.scm*
und
aktualisiere-besuchliste_Erarbeitung.scm → *aktualisiere-besuchliste_Kurs.scm*
- *aktualisere-kuerzeste-wege_Erarbeitung.scm* gelingt nicht mehr.
- Hinweis auf nachträglich erstellte Präsentationen zu den Aufgaben.