

# Pointer-Listen - eine Anwendungsaufgabe

## Sonderübung

Nadine Schärmann & Thomas Klütz

PROG

4. Mai 2020

# Organisatorisches

- Blatt9 bis 08.05.2020, Abgabe im Abgabebriefkasten im OPAL
- Fragen zur Übung im Matrix-Chatraum
- Lösungen der letzten Sonderübung im gitlab

Gebt uns gerne ein Feedback!

# Letzte Woche

Was haben wir letzte Woche gemacht?

- Grundfunktionen von Pointer-Listen am Beispiel einer Einkaufsliste
- Einfügen von Elementen in eine alphabetisch-sortierte Liste

# Aufgabe: Dünn besetzte Vektoren

Ein Vektor  $v$  heißt dünn besetzt, wenn die meisten Komponenten gleich Null sind. Ein solcher Vektor kann als verkettete Liste von Elementen mit positivem Index dargestellt werden.

Definiert dafür ein Modul mit:

- einem Datentyp `content` mit einer ganzzahligen Komponente für den Index und einer reellen Komponente für den Wert  $v_i$ , der  $i$ -ten Komponente eines spärlich besetzten Vektors. (Die Initialisierung der Datenstruktur muss in der Typdefinition erfolgen.)
- einem Datentyp `node` für ein Listenelement mit einer Komponente des Typs `content` und einer Zeigerkomponente, die auf das nächste Element der Liste zeigt.
- einem Datentyp `sparvektor` mit einer Zeiger-Komponente des Typs `node`, die auf das erste Listenelement des Vektors zeigt.

# Aufgabe: Dünn besetzte Vektoren

Weiterhin sollen im Modul

- eine Subroutine `lesevektor`, die so lange Paare  $(i, v_i)$  einließt und in einem neu zu erzeugenden Vektor einträgt, bis ein Paar mit negativem Index eingelesen wird. (Die einzulesenden Paare sind nach aufsteigendem Index geordnet.)
- einen Operator `*` mit entsprechend implementierter Funktion zur Berechnung des Skalarproduktes zweier dünn besetzter Vektoren.

definiert werden.