

Praxiseinheit 2 - Freispeicherverwaltung & ADT

1. Freispeicherverwaltung

Implementieren Sie selbst eine Freispeicherverwaltung für eine dynamische Datenstruktur `struct intElem { int info; intElem *pred, *succ; };`

Als Freispeicher steht ein Array mit 1000 Elementen vom Typ `intElem` zur Verfügung.

- a) Schreiben Sie eine Funktion `request()`, die einen Zeiger auf ein freies Element des Arrays zurückliefert oder den Nullzeiger zurückliefert, wenn kein freier Speicher mehr zur Verfügung steht!
- b) Schreiben Sie eine Funktion `free(intElem* el)`, die den Speicher für `el` wieder freigibt!

2. Entwickeln eines ADT `intList`

Kapseln Sie Funktionalität und Datenstruktur von 1. in einem ADT und ergänzen Sie:

- eine Methode zum geordneten Einfügen einer ganzen Zahl
- eine Methode zum Test auf die Existenz einer gegebenen ganzen Zahl
- eine Methode zum Entfernen einer gegebenen ganzen Zahl aus der Datenstruktur

3. Anwendung des ADT `intList`

- Erzeugen Sie (algorithmisch) eine Beispielliste, die alle ganzen Zahlen von 2 bis 50 enthält! Entfernen Sie dann (algorithmisch) jede gerade Zahl aus der Liste, danach jede durch 3 teilbare und zuletzt jede durch 5 teilbare Zahl!
- Zeichnen Sie (manuell) die verbliebene Listenstruktur und erläutern Sie deren Werte!