

SWE3 HÜ 6: Iteration

Erik Pitzer <erik.pitzer@fh-hagenberg.at>

Medizin- und Bio-Informatik – WS 2024/25

Name

Aufwand (in h)

Punkte

Aufgabe	Lösungsidee	Implementierung	Tests	Gesamtpunkte
1 a	20% / 7P	50% / 18P	30% / 10P	35
1 b	20% / 6P	50% / 15P	30% / 9P	30
1 c	20% / 5P	50% / 13P	30% / 5P	25
1 d	10P			10

Erstellen Sie für die folgenden Aufgaben eine gemeinsame Solution und die jeweilige Aufgabe als Projekt in einem Unterverzeichnis.

Aufgabe 1: Liste mit Iterator

Entwickeln Sie analog zum Vektor aus der Übung einen weiteren Container der eine Iteration bietet. Implementieren Sie dazu eine doppelt verkettete lineare Liste die einen bidirektionalen Iterator zur Verfügung stellt. Die Liste sollte dabei wieder Templates unterstützen und einen beliebigen Element-Typ erlauben. Dokumentieren Sie die Struktur der Liste und gehen Sie vor allem auf die Implementierung des Iterators ein.

- a) Stellen Sie für folgende Anforderungen jeweils passende Methoden zur Verfügung.
- Eine Möglichkeit eine leere Liste zu erstellen.
 - Eine (effiziente!) Möglichkeit am Anfang (`push_front`) und am Ende (`push_back`) der Liste Elemente einzufügen.
 - Die Länge der Liste abfragen (`size`).
 - Elemente in der Liste finden (`find`).
- b) Sorgen Sie außerdem dafür, dass die Liste iterierbar ist (`begin` und `end`) und implementieren Sie dafür einen passenden Iterator. Überlegen und dokumentieren Sie, welche Art von Iterator für Ihre Liste passend ist und versehen Sie den Iterator mit dem passenden `iterator_tag`.
- c) Implementieren Sie ein Methode `foreach` die mit Hilfe eines Funktors oder Funktionszeigers beliebige Funktionen auf jedes Element anwenden kann.
- d) Überlegen Sie ob es in Ihrer Implementierung möglich wäre die Liste während der Iteration zu verändern – insbesondere ein Element zu löschen. Machen Sie ein Design-Skizze wie Sie diese Funktionalität implementieren *könnten*.