

1 ArrayList

1.1 Einleitung

Ein ArrayList implementiert die Liste Interface. ArrayList unterstützt dynamische Arrays, die ihrer Größe sich ändert wenn nötig.

Normale Arrays haben eine bestimmte Größe, wenn sie erzeugt sind dann dieser Größe kann nicht verändert werden. Die haben die Nachteil, dass die Anzahl der Elementen sollen schon bekannt sein.

ArrayLists löst dieser Problem, bei die automatische Größe Veränderung von den Array. Wenn ein ArrayList schon voll ist, und ein neues Element eingefügt ist, wird die ArrayList automatisch die Größe erhöhen, sodass es Platz für die neues Element gibt. Wenn Elementen gelöscht sind, es ist möglich der Array Größe zu verkleinern.

In ArrayList gibt es drei Konstruktoren:

- ArrayList() Erzeugt eine leere array Liste
- ArrayList(Collection c) Erzeugt eine array Liste mit den Elementen von dem Collection c
- ArrayList(int capacity) Erzeugt eine array Liste mit gewünschte Kapazität. Diese Kapazität wurde automatisch größer wenn die array Liste die Kapazität erreicht.

1.2 Methodenbeschreibung

- add(Object o) - addiert ein Element zu die Liste.
- add(int index, Object o) - addiert der Element in den eingegebene Stelle.
- addAll(Collection c) - addiert alle Elementen in einer Collection.
- addAll(int index, Collection c) - addiert alle Elementen ab die eingegebene Stelle.
- clear() - löscht alle Elementen in der Liste.
- contains(Object o) - gibt boolean Wert ob die gefragte Element in der Liste ist.
- get(int index) - gibt den Element zurück an der eingegebene Stelle.
- remove(int index) - Löscht der Element in der eingegebene Stelle.
- set(int index, Object o) - Ersetzt der Element in der eingegebene Stelle mit einem neuen Element

- `size()` - gibt die gröÙe der Liste zurück
- `Object[] toArray()` - erzeugt ein Array mit die Elementen der Liste

1.3 Effizienzanalyse

Graph einfügen und Effizienz ausrechnen. Für einzelne Probleme anwenden.

1.4 Schlussfolgerung

hier die resultierenden Stärken und Schwächen rausschreiben.