

1 Datastructuren

1.1 Stack

Definitie: een stapel / stack is een datastructuur waarbij de toegang to de elementen beperkt wordt tot het meest recent toegevoegde element.

- `s.push(x)` = element `x` toevoegen aan stack `s`. toegevoegde element is de nieuwe top van de stapel
- `s.pop()` = bovenste element verwijderen en geeft het terug
- `s.peek()` = toont bovenste element

1.2 Queue

Definitie: een wachtlijn / queue is een datastructuur waarbij de toegang to de elementen beperkt wordt tot het minst recent toegevoegde element.

- `q.enqueue(x)` = element `x` toevoegen aan de staart van de queue `q`
- `q.dequeue()` = haalt element aan de kop weg uit de wachtlijn `q` en geeft het terug
- `q.front()` = geeft voorste element van aan de kop van de wachtlijn

1.3 Collections

Definitie: een collection / container is een object dat meerdere elementen tot 1 geheel groepeert.

- Basisbewerkingen
 - `c.size()` = aantal elementen van `c`
 - `c.isEmpty()` = leeg?
 - `c.contains(x)` = zoekfunctie
 - `c.add(x)` = toevoegen
 - `c.remove(x)` = verwijderen
- Bulkbewerkingen
 - `c1.containsAll(c2)` = checkt of `c1` alle elementen van `c2` bevat
 - `c1.addAll(c2)` = voegt alle elementen van `c2` aan `c1` toe
 - `c1.removeAll(c2)` = verwijdert alle elementen van `c2` die in `c1` zitten
 - `c1.retainAll(c2)` = verwijdert uit `c1` alle elementen die NIET ook in `c2` zitten
 - `c.clear()` = clearen
- Arraybewerkingen
 - `c.toArray()` = maakt een `Object[]` array van de collection `c`

- `c.toArray(a)` = maakt nieuwe array waarbij type bepaald wordt door de componenten

- Iterator

- Iterator `it = c.iterator()`;

1.4 List

Definitie: is een geordende collection die dubbels kan bevatten

- Basisbewerkingen

- `l.remove(x)` = item deleten uit de lijst
- `l.add(x)` = item toevoegen aan lijst
- `l.addAll(c)` = voegt items van de collection toe aan de lijst

- Positie

- `l.get(p)` = geeft item uit l op plaats p terug
- `l.set(p,x)` = vervangt in p door x in l
- `l.add(p,x)` = x toevoegen in de lijst op plaats p
- `l.remove(p)` = element op plaats p verwijderen
- `l.addAll(p,c)` = items uit c toevoegen aan de l op plaats p

- Listiterator

- `l.listIterator()` = begin aan begin van l
- `l.listIterator(p)` = gaat naar plaats p van de lijst

- Range-view

- `l.subList(p,q)` = maakt sublijst van p tot q van l

- Algoritmen

- `Collections.sort(l)` = lijst sorteren volgens merge-sort
- `Collections.reverse(l)` = omkeren van de lijst

1.5 Set

Definitie: is een geordende collection die geen dubbels kan bevatten

- Basisbewerkingen

- `s.size()` = aantal elementen van c
- `s.isEmpty()` = leeg?
- `s.contains(x)` = zoekfunctie
- `s.add(x)` = toevoegen
- `s.remove(x)` = verwijderen
- `s.iterator()` = Iterator over Set s

- Bulkbewerkingen
 - `s1.containsAll(s2)` = checkt of s1 alle elementen van s2 bevat
 - `s1.addAll(s2)` = maakt unie van s1 en s2 -> s1
 - `s1.removeAll(s2)` = pakt verschil van s1 en s2 -> s1
 - `s1.retainAll(s2)` = doorsnede s1 en s2 -> s1
 - `c.clear()` = clearen
- Arraybewerkingen
 - `c.toArray()` = geeft een array van de set s
 - `c.toArray(a)` = maakt nieuwe array waarbij type bepaald wordt door de componenten

1.6 SortedSet

Definitie: is een set die zijn elementen in stijgende volgorde bijhoudt volgens natuurlijke ordening

- Eindpunt bewerkingen
 - `s.first()` = kleinste element van s
 - `s.last()` = grootste element van s
- Range-view
 - `s.subSet(x1,x2)` = subset van x1 tot x2
 - `s.headSet(x2)` = subset van elementen die strikt kleiner zijn dan x2
 - `s.tailSet(x1)` = subset van elementen groter of gelijk aan x1

1.7 Map

Definitie: een map is een object dat sleutels op waarden afbeeld. Er kunnen geen duplicaten aanwezig zijn.

- Basisbewerkingen
 - `m.put(k,w)` = hangt de waarde w aan k
 - `m.get(k)` = geeft waarde van k terug
 - `m.remove(k)` = verwijdert k
 - `m.containsKey(k)` = zoeken naar een sleutel k
 - `m.containsValue(w)` = zoeken naar waarde w
 - `m.size()` = aantal entries in m
 - `m.isEmpty()` = leeg?
- Bulkbewerkingen
 - `m.clear()` = leegmaken
 - `m.putAll(m2)` = mappings van m2 kopiëren in m

- Collection-views
 - `m.keySet()` = Set van alle sleutels van `m`
 - `m.values()` = Collection an alle waarden van `m`
 - `m.entrySet()` = Set van alles van `m`

1.8 SortedMap

Definitie: een map die zijn entries in stijgende volgorde bijhoudt, gesorteerd volgende natuurlijke volgorde van de sleutels. Neemt de functies over van een `Map` + `Iterator` en `toArray()`