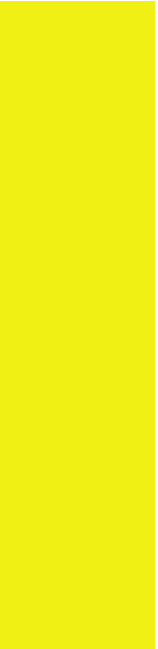


01 Collections - Nachbesprechung

Algorithmen und Datenstrukturen 2



SortedBag - SortedSet - UnsortedBag - UnsortedSet

Gegeben sind mehrere Anwendungsfälle. Entscheiden Sie sich für die passendste der vier Datenstrukturen und *begründen* Sie Ihren Entscheid.

SortedBag - SortedSet - UnsortedBag - UnsortedSet

Eine Werbeagentur sammelt E-Mail Adressen, an welche Sie dann Spammails versendet. Welche Datenstruktur empfiehlt sich für die E-Mail Sammlung?

SortedBag - SortedSet - UnsortedBag - UnsortedSet

Eine Werbeagentur sammelt E-Mail Adressen, an welche Sie dann Spammails versendet. Welche Datenstruktur empfiehlt sich für die E-Mail Sammlung?

Da jede E-Mail Adresse nur 1x vorkommen soll wird eine **Set-Struktur** benötigt. Da jede Einfüge-Operation eine Suchoperation benötigt, was bei sortierten Daten schneller geht, empfiehlt sich die **SortedSet** Variante.

Eine andere Möglichkeit wäre das **UnsortedSet**, da Einfüge-Operationen sowieso in $O(n)$ liegen und hier hauptsächlich eingefügt wird sowie am Schluss über die komplette Liste iteriert wird.

Eine dritte Möglichkeit wäre die Verwendung eines **UnsortedBags** mit der Begründung, dass schnell eingefügt werden soll und es egal ist, ob ein Empfänger die E-Mails mehrmals empfängt.

SortedBag - SortedSet - UnsortedBag - UnsortedSet

Sie entwerfen Software für eine Bibliothek, wo alle Bücher abgespeichert werden. Wenn ein Buch in mehreren Exemplaren vorhanden ist, wird es auch mehrmals abgelegt (weil die Exemplare einzeln ausgeliehen werden können). Die Kunden der Bibliothek nutzen das System für die Suche (nach Titel).

SortedBag - SortedSet - UnsortedBag - UnsortedSet

Sie entwerfen Software für eine Bibliothek, wo alle Bücher abgespeichert werden. Wenn ein Buch in mehreren Exemplaren vorhanden ist, wird es auch mehrmals abgelegt (weil die Exemplare einzeln ausgeliehen werden können). Die Kunden der Bibliothek nutzen das System für die Suche (nach Titel).

Ein Buch kann auch mehrfach vorkommen: Bag-Semantik: Die Kunden führen viele Suchanfragen durch:
Sortierte Datenablage => **SortedBag**

SortedBag - SortedSet - UnsortedBag - UnsortedSet

Die Kartbahn Wohlen möchte eine Statistik über die Rundenzeiten der Fahrer machen. Sie möchte am Ende des Tages die schnellste und langsamste Rundenzeit auslesen.

SortedBag - SortedSet - UnsortedBag - UnsortedSet

Die Kartbahn Wohlen möchte eine Statistik über die Rundenzeiten der Fahrer machen. Sie möchte am Ende des Tages die schnellste und langsamste Rundenzeit auslesen.

Da die genau gleiche Rundenzeit öfters vorkommen kann muss eine **Bag-Semantik** Collection verwendet werden. Da die Rundenzeiten nur am Ende des Tages ausgewertet werden und während des ganzen Tages Zeiten eingefügt werden empfiehlt sich eine Datenstruktur, die schnell Einfügeoperationen durchführen kann. Die Reihenfolge ist weniger wichtig. Am Ende des Tages braucht die Suche für die schnellste und langsamste Rundenzeit noch eine Iteration mit $O(n)$. Deshalb empfiehlt sich hier **UnsortedBag**.