



Fachbereich Informatik

Prof. Dr. Peter Knauber

Bachelor-Studiengang

Übungen zur Vorlesung Grundlagen der Informatik, WS 2004/2005

Übungsblatt 12: Abstrakte Datentypen in Java Ausgabe am: 11.1.2005

Abgabe am: Freitag, 21.1.2005

Betrachtet wird ein abstrakter Datentyp *Date*:

Der ADT *Date* repräsentiert ein bestimmtes Datum im Gregorianischen Kalender und bietet verschiedene Operationen auf Kalenderdaten an.

Die jeweiligen Operationen sind wie folgt definiert:

- *Date*: $\text{int} \times \text{int} \times \text{int} \rightarrow \text{Date}$
Wenn es sich um ein gültiges Datum handelt, liefert dieser Konstruktor ein entsprechendes *Date*-Objekt als Ergebnis; sonst wird ein *Date*-Objekt geliefert, für das *getDay()*, *getMonth()* und *getYear()* jeweils -1 als Ergebnis liefern
- *equals*: $\text{Date} \times \text{Date} \rightarrow \text{boolean}$
Liefert genau dann *true*, wenn die beiden Kalenderdaten gleich sind
- *laterThan*: $\text{Date} \times \text{Date} \rightarrow \text{boolean}$
Liefert genau dann *true*, wenn das 1. Datum später als das 2. Datum ist
- *earlierThan*: $\text{Date} \times \text{Date} \rightarrow \text{boolean}$
Liefert genau dann *true*, wenn das 1. Datum früher als das 2. Datum ist
- *getWeekday*: $\text{Date} \rightarrow \text{int}$
Liefert den Wochentag zu einem Datum;
der Montag entspricht der 1, der Dienstag der 2 usw.
- *subtract*: $\text{Date} \times \text{Date} \rightarrow \text{int}$
Subtrahiert das 2. Datum vom 1. Datum; liefert die Differenz der beiden Datumsangaben in Tagen als ganze Zahl
Voraussetzung: das 1. Datum ist größer als das 2. Datum, sonst ist das Ergebnis -1
- *isValid*: $\text{int} \times \text{int} \times \text{int} \rightarrow \text{boolean}$
Liefert genau dann *true*, wenn die angegebenen Zahlen in der Reihenfolge Tag, Monat, Jahr ein gültiges Datum repräsentieren
- *getDay*: $\text{Date} \rightarrow \text{int}$
getMonth: $\text{Date} \rightarrow \text{int}$
getYear: $\text{Date} \rightarrow \text{int}$
Liefern den Tag, den Monat bzw. das Jahr des angegebenen Datums als *int*-Wert

Die verschiedenen Methoden (nicht jedoch der Konstruktor, die *get*-Methoden und *isValid*!) dürfen sich darauf verlassen, dass sie nur mit gültigen Datumsangaben aufgerufen werden.

Aufgabe 1

10 Punkte

Informieren Sie sich über den Gregorianischen Kalender. Beschreiben Sie kurz, wie dieser definiert ist.

Aufgabe 2

79 Punkte

Implementieren Sie eine Klasse *Date* in Java, welche alle oben angegebenen Methoden außer *isValid* als **Instanzmethoden** enthält; die Methode *isValid* ist als **statische Methode** zu implementieren. Sie dürfen beliebige Attribute und zusätzliche eigene Klassen oder Methoden, aber keinerlei Hilfsmethoden aus der Java-Bibliothek verwenden.

Testen Sie Ihre Klasse gründlich, zum Beispiel mit Hilfe einer zusätzlichen Testklasse.

Hinweise

- Es wird ein automatisches Testprogramm für Aufgabe 2 geben; dafür muss auf jeden Fall der geforderte Konstruktor implementiert werden.
- Überlegen Sie sich vor der Bearbeitung dieser Übungsaufgaben, wie viele Punkte Sie noch benötigen, um die 75%-Klausurzulassungsquote zu erreichen und wie viele (Teil-)Aufgaben Sie bearbeiten sollten, um dieses Ziel **sicher** zu erreichen!
Es ist vermutlich nicht notwendig, alle (Teil-)Aufgaben zu bearbeiten; für das Abschneiden in der Klausur ist nicht relevant, ob Sie zum Beispiel 75% oder 98% der Übungspunkte erreicht haben.