Übungen zu Vorlesung
Praktische Informatik I
Wintersemester 2002/03

Serie 3

Mengen, Bitvektoren und Arrays

Aufgabe 1: „Kleine“ Mengen als Bitvektoren (60%)

32–Bit Integer Zahlen eignen sich, um Mengen mit maximal 32 verschiedenen Elementen zu repräsentieren, indem jeweils ein Bit verwendet wird, um zu markieren, ob ein Wert in der Menge enthalten ist oder nicht. Die üblichen Operationen auf Mengen, wie zum Beispiel Vereinigung, Schnittmengenbildung und Differenzmengenbildung, lassen sich dann in Java mit Hilfe der Bitoperatoren realisieren.
Auf der PI1 Webseite steht in dem File `sets.java` ein Programm–Skelett zur Verfügung, welches erweitert werden soll. Schreiben Sie zu jedem Kommentar in der genannten Datei ein Stück Java-Code, welches die entsprechenden Operationen auf Mengen (bei Verwendung der oben beschriebenen Umsetzung mit Bitvektoren) implementiert.
Geben Sie zu jeder Operation *precondition/postcondition* Paare an, welche die Operation spezifizieren. Dabei soll die in der Vorlesung besprochene Abstraktionsfunktion $\mathcal{A}$ verwendet werden.

Aufgabe 2: „Größere“ Mengen mit Arrays (40%)

Das Konzept, jeweils ein Bit zur Speicherung des Enthaltenseins eines Elements einer Menge zu verwenden, läßt sich durch die Verwendung eines *Arrays* von 32–Bit Zahlen (statt nur einer Variablen) auf beliebig große endliche Mengen erweitern. Wenn z. B. ein Array namens `set_array` zum Speichern einer Menge verwendet wird, so läßt sich das 33. Element in dem 0. Bit der Variablen `set_array[1]` speichern.
Legen Sie ein Array an, um Mengen mit `MAX_ELEM` verschiedenen Elementen handhaben zu können, wobei `MAX_ELEM` eine Konstante sein soll. Integrieren Sie ebenfalls, vergleichbar zu dem Skelett in Aufgabe 1, Aufrufe der Funktionen, um sie mit sinnvollen und interessanten Testfällen auszuprobieren.

Abgabe: 3. – 6.12. nach den jeweiligen Praktika. Die Abgabe soll sowohl elektronisch (Programm-Quellcode) als auch in gedruckter Form (mit LaTeX gesetzter kommentierter und erläuterter Quellcode) erfolgen!