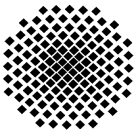
Universität Stuttgart
Fakultät 5, Institut FMI
EfidI 1, Professor Claus



WS 2002/2003
Bertele, Lewandowski
13. Januar 2003

Aufgabenblatt 10

Abgabe bis 9.1.2003, 20:00 Uhr – Besprechung in der Woche ab dem 13.1.2003

Aufgabe 1 Mengen I (schriftlich, mittel) 5 Punkte

In dieser Aufgabe üben Sie sowohl die Erzeugung eines ADT, sowie den Umgang mit Listen.

Implementieren Sie einen abstrakten Datentyp IntSet, der eine Menge über dem gesamten Wertebereich des Datentyps Integer realisiert und auf dem folgende Operationen zur Verfügung gestellt werden:

- `Create(S: in out IntSet);` Erzeugt eine leere Menge.
- `Copy(S: IntSet) return IntSet;` Liefert eine Kopie der Menge S.
- `Incl(S: in out IntSet; E: Integer);` Nimmt das Element E in die Menge S auf.
- `Excl(S: in out IntSet; E: Integer);` Entfernt das Element E aus der Menge S.
- `IsIn(S: IntSet; E: Integer) return Boolean;` Liefert True, falls das Element E in S vorkommt, sonst False.

Verwenden Sie zur Repräsentation einer Menge eine verkettete Liste, deren Elemente der Größe nach sortiert sind; diese Eigenschaft soll nach jeder Operation erhalten bleiben.

Aufgabe 2 Mengen II (schriftlich, schwer) 8 Punkte

In dieser Aufgabe geht es um effiziente Programmierung.

Implementieren Sie weitere Operationen auf Ihrem abstrakten Datentyp IntSet aus „Mengen I“:

- `Difference(S1, S2: IntSet) return IntSet;`
  Liefert die Mengendifferenz $S1 - S2 = \{x \mid x \in S1 \wedge x \notin S2\}$.
- `Intersection(S1, S2: IntSet) return IntSet;`
  Liefert die Schnittmenge $S1 \cap S2$.
- `SymmetricDifference(S1, S2: IntSet) return IntSet;`
  Liefert die symmetrische Differenz $\{x \mid (x \in S1 \wedge x \notin S2) \vee (x \in S2 \wedge x \notin S1)\}$.
- `Union(S1, S2: IntSet) return IntSet;`
  Liefert die Vereinigungsmenge $S1 \cup S2$.

Ihre Lösung soll die Sortierung der Listenrepräsentation in aufsteigender Reihenfolge erhalten und ausnutzen. Sie können zur Implementierung die Prozedur Copy aus „Mengen I“ verwenden, aber nicht die Prozeduren Incl, Excl und In. (Dadurch sollten Sie lineare Komplexität erreichen.)

Aufgabe 3 Exceptions (Votieraufgabe, mittel) 7 Punkte

Diese Aufgabe fasst das Arbeiten mit Arrays und Exceptions zusammen. Außerdem sollen Sie sich in dieser Aufgabe auch Gedanken machen, wie Programme getestet werden können.

Schreiben Sie ein Paket „Felder“ (ADS und ADB) zur Verarbeitung von unbegrenzten Zeichenketten. Definieren Sie dazu einen Typ, der ein Array aus Ustrings darstellt. Schreiben Sie dann folgende Routinen:

- `function erzeuge(Start, Ende: Integer) return Feldtyp;`
  Liefert ein Array mit den angegebenen Grenzen zurück. Die Elemente sollten leer sein.
- `function erzeuge(Anzahl: Integer) return Feldtyp;`
  Wie die vorige Funktion, aber hier wird ein Array mit der angegebenen Anzahl von Elementen zurückgeliefert.
- `function hole(Feld: Feldtyp; Position: Integer) return Ustring;`
  Liefert den an der Stelle „Position“ gespeicherten String zurück. Ist der String leer, wird die Exception „Leer“ geworfen.
- `procedure setze(Feld: in out Feldtyp; Position: in Integer; Wert: Ustring);`
  Trägt an der Stelle „Position“ den String „Wert“ ein. Ist an dieser Stelle bereits an String eingetragen, wird die Exception „Voll“ geworfen.
- `function frei(Feld: feldtyp; Position: Integer) return Boolean;`
  Überprüft, ob die Stelle „Position“ noch frei ist.
- `procedure loesche(Feld: in out Feldtyp; Position: in Integer);`
  Löscht die entsprechende Stelle im übergebenen Feld. Ist die Stelle bereits leer, wird die Exception „leer“ geworfen.
- `function zaehle(Feld: Feldtyp) return Natural;`
  Zählt die Anzahl der im Feld enthaltenen, belegten Strings.

Schreiben Sie zusätzlich noch ein Hauptprogramm, das ihr Programmpaket testet. Das heißt es sollen sowohl funktionierende Fälle, als auch Fehlerfälle überprüft werden. Das heißt, sie sollten die Exceptions im Hauptprogramm abfangen. Bedenken Sie, dass außer den von Ihnen erzeugten Exceptions auch noch die Ada95-Exceptions geworfen werden können.

Allgemeine Hinweise:

- Die schriftlichen Aufgaben (13 Punkte) geben Sie bitte zum Abgabezeitpunkt im eClaus-System ab. Bitte votieren Sie bitte ebenfalls bis zum Abgabezeitpunkt im eClaus-System.
- Bei weiteren Fragen, wenden Sie sich bitte an das Forum (http://fachschaft.informatik.uni-stuttgart.de/forum/), Ihren Tutor, oder per Mail direkt an J. Bertele (inf@studbs.informatik.uni-stuttgart.de).

Weitere Hinweise finden Sie auf unserer Veranstaltungswebseite unter:
http://www.informatik.uni-stuttgart.de/ifi/fk/lehre/ws02-03/info_I_0203.html