Übungsblatt 12 zu "Programmiersprachen"

Berthold Hoffmann, Studiengang Informatik (hof@informatik.uni-bremen.de)

Besprechung am 1. Juni 2010

Generische Abstraktion

Betrachten Sie ein generische Paket für Listen über einem Typparameter Element.

Die Paketsachnittstelle ist:

```
generic
  type Element is private;
package Lists is
  type List is limited private;
  -- A List value represents a list whose elements are of type Element.
  procedure clear (l: out List);
  -- Make list l empty.
  procedure add (l: in out List; e: in Element);
  -- Add element e to the end of list l.
  . . . -- other operations
private
  capacity: constant Integer := . . .;
  type List is
          record
             length: Integer range 0 .. capacity;
                                             7.2 Type and class parameters
            elems: array (1 .. capacity) of Element;
         end record;
end Lists;
```

Der Rumpf des Pakets ist

```
package body Lists is
  procedure clear (l: out List) is
  begin
        l.length := 0;
  end;
  procedure add (l: in out List; e: in Element) is
  begin
        l.length := l.length + 1;
        l.elems(l.length) := e;
  end;
  . . . -- other operations
end Lists;
```

Dann können Pakete so instantiiert werden:

```
package Phrases is new Lists(Character);

 ...

type Transaction is record . . . end record;
package Transaction_Lists is new Lists(Transaction);
```

Definieren Sie dieses Paket in einer der Sprachen

- Java
- C++
- C#