Grundkurs Common-Lisp 12. (freiwillige) Übung (12-FEB-1998)

1 Auswertungsstrategie (5 Punkte)

Erkläre die Auswertungsstrategie des COMMON-LISP Interpreters für die folgenden Ausdrücke: (a) Zahlen, (b) Zeichenketten (*Strings*), (c) Symbole, (d) Schlüßelwortsymbole (*key words*) und (e) Listen. Beachte die Unterscheidung zwischen Funktionen und *special forms* oder Makros; gib je mindestens ein Beispiel an.

2 Makros (5 Punkte)

Gegeben sei die folgende Definition für ein zweistelliges Makro `foo()`:

```
(defmacro foo (bar baz)
  `(setq ,baz (cons ,bar ,baz)))
```

(a) Wozu expandiert der Makroaufruf '`(foo 42 fee)`'?

(b) Welches vordefinierte COMMON-LISP Makro entspricht in seiner Funktionsweise `foo()`?

(c) Warum läßt sich `foo()` nicht als Funktion definieren?

3 Iteration (5 Punkte)

Schreibe eine zweistellige iterative Funktion `vereinige()`, die aus zwei Mengen in Listendarstellung (jedes Element ist nur einmal enthalten; die Ordung der Elemente ist irrevelant) die Vereinigungsmenge; e.g.

```
? (vereinige '(a b c) '(d e a))
→ (C B D E A)
```

4 Listenzugriffe (1 + 1 + 1 + 1 Punkte)

Die globale Variable `*foo*` sei wie folgt belegt:

```
(defvar *foo* '(achim (birgit) chris (doris (eva) (fritz))))
```

Wozu evaluieren die Zugriffsausdrücke:

(a) `(first (first (rest *foo*)))`

(b) `(first (rest (fourth *foo*)))`

(c) `(rest (first (rest (nth 3 *foo*))))`

(d) `` `(adam ,(first (first (rest (rest (nthcdr 3 *foo*)))))) ``

5 Listenkonstruktion (1 + 1 + 1 + 1 Punkte)

Es gelten die globalen Bindungen

```
(defvar x '(1 4 5))
(defvar y '(2 3 6))
```

Wozu evaluieren die folgenden Konstruktionsausdrücke

(a) `(cons 4 (cons 7 (cons 1 (cons 1 (cons nil)))))`

(b) `(cons x y)`

(c) `(append (cons 'x y) (cons 'y x))`

(d)
```
(list (cons (first x) (cons (second y) (rest (rest x))))
      (cons (first y) (cons (second x) (rest (rest y)))))
```

6 Wir spielen Interpreter (5 Punkte)

Beschreibe möglichst genau die Funktionsweise der folgenden Funktion:

```
(defun ? (foo bar)
  (if (null foo)
    bar
    (cons (first foo) (? (rest foo) bar))))
```

Gibt es eine vordefinierte COMMON-LISP Funktion, die in ihrem Verhalten `?()` entspricht?

Klausurdauer: 90 Minuten