

Aufgabe 1 (6 Pkt):

Die Parameterübergabe-Variante call-by-reference bringt mit sich, daß jede eine Reference-Variable betreffende, ihren Inhalt modifizierende Zuweisung nach außen sichtbar wird. Dies könnte zur Folge haben, daß "unzulässige" Wertdarstellungen sichtbar werden (einen geteilten Speicher vorausgesetzt). Die Ersetzung des call-by-reference-Parameterübergabe-Mechanismus durch **call-by-copy/restore** kann dies verhindern ("restore" als atomar vorausgesetzt). Die Wirkung kann wie folgt beschrieben werden: der Inhalt der Speicherstelle, auf die der aktuelle Reference-Parameter zeigt, wird in eine lokal aufgefaßte Variable kopiert. Nach Prozedur-Abarbeitung, agierend mit der "Kopie" anstatt dem "Original", wird dann der Inhalt der Speicherstelle, auf die der Original-Reference-Parameter zeigt, überschrieben.

Fernaufrufe von Prozeduren, die nur den call-by-value-Parameterübergabe-Mechanismus verwenden, sind recht problemlos und transparent zu realisieren. Sieht die Situation genauso problemlos aus, wenn der Fernaufruf Prozeduren betrifft, die auch den call-by-reference-Parameterübergabe-Mechanismus verwenden, wobei der Fernaufruf den call-by-copy/restore-Mechanismus nutzt, um den call-by-reference-Parameterübergabe-Mechanismus "nachzubilden" ?

Argumentieren Sie anhand des folgenden Beispiel-Programms, wobei die Prozedur "DoppelInkrement" durch Fernaufruf (unter Nutzung des call-by-copy/restore-Mechanismus) zu erreichen sei:

```
procedure DoppelInkrement (var a,b: integer);  
begin  
  a := a + 1;  
  b := b + 1;  
end;
```

```
program test;  
var x: integer;  
{ "Fern"-Prozedur: procedure DoppelInkrement(var a,b: integer)};  
begin  
  x := 0;  
  DoppelInkrement(x,x);  
  writeln(x);  
end.
```

Aufgabe 2 (6 Pkt):

Stellen Sie ein Flugbuchungssystem, das mehrere Reisebüros unterstützt, als Client-Server-System dar. Geben Sie auch benötigte Operationen an.

Aufgabe 3 (3 Pkt):

Ordnen Sie den durch die Paradigmen,

- i) Clients und Server,
- ii) Datenbank und kooperierende Prozesse und
- iii) gemeinsame Aktionen und kooperierende Prozesse,

eingeführte Komponententypen folgende Komponenteneigenschaften sinnvoll zu:

- a) aktiv \leftrightarrow passiv
- b) temporär vorhanden \leftrightarrow permanent vorhanden
- c) lokal bekannt \leftrightarrow global bekannt
- d) lokal zugreifbar \leftrightarrow global zugreifbar
- e) stimulierende Funktionalität \leftrightarrow reagierende Funktionalität.

Ihre Lösungen können Sie bis Mittwoch, 7.2.2001, 12 Uhr in die Briefkästen im Pav. 6 einwerfen.