

Blatt 10

Wichtige Hinweise:

- Alle Programmieraufgaben müssen in doppelter Form abgegeben werden. Einerseits als Ausdruck oder handschriftlich in den Übungen und außerdem per E-Mail an die Adresse:

`informatik-ueb@informatik.uni-giessen.de`

Im Betreff **muß** das Wort “Hausaufgabe” stehen; ansonsten wird die E-Mail nicht angenommen.

- Programme müssen kommentiert sein.
- Zum Lösen der Aufgaben dürfen nur Programmierelemente benutzt werden, die im Skript einschließlich Kapitel 10 behandelt wurden.

(10.1) [8 Punkte]

Schreiben Sie ein Pascalprogramm, das eine einfach verkettete Liste bestehend aus den folgenden Elementen verarbeiten kann.

```
TYPE ZeigEle = ^Element;  
Element = RECORD  
    Zahl: integer;  
    Next: ZeigEle;  
END;
```

1. Schreiben Sie eine Prozedur **einfuegen**, die den Anker einer einfach verketteten Liste und eine ganze Zahl x übergeben bekommt. Es soll nun ein neues Element mit der Zahl x erzeugt werden und an das Ende der Liste eingefügt werden.
2. Schreiben Sie eine Prozedur **ausgeben**, die den Anker einer einfach verketteten Liste übergeben bekommt und die Zahlen der Liste geeignet ausgibt.

3. Schreiben Sie eine Funktion **einfuegen_vor**, die den Anker einer einfach verketteten Liste und zwei ganze Zahlen x und y übergeben bekommt. Es soll nun ein neues Element mit der Zahl x erzeugt werden und vor dem ersten Auftreten von y in die Liste eingefügt werden, falls ein Eintrag mit y existiert. In diesem Fall soll die Funktion **true** und ansonsten **false** zurückgeben.

Betten Sie die Definition der einfach verketteten Liste, die beiden Prozeduren sowie die Funktion in ein Pascalprogramm ein und testen Sie in dem Programm die Funktion **einfuegen_vor** sowie die beiden Prozeduren. Denken Sie daran, Ihr Programm geeignet zu kommentieren.

(10.2) [6 Punkte]

Betrachten Sie das folgende Programm. Was wird ausgegeben und warum? Bitte geben Sie eine genaue Begründung.

```
program test;  
  
type pinteger=^integer;  
var i,j,k : pinteger;  
  
procedure eins(var i:integer; j,k:integer);  
begin  
    i:=1; j:=1; k:=1;  
end;  
  
procedure zwei(i,j,k:pinteger);  
  
    procedure drei(i:integer; var j,k:integer);  
    begin  
        i:=3; j:=3; k:=3;  
    end;  
  
begin  
    drei(k^,i^,j^);  
end;  
procedure vier(j:integer; var k:integer);  
var i: integer;  
begin  
    i:=4; j:=4; k:=4;  
end;
```

```

begin
    new(i); new(j); new(k);
    i^:=9; j^:=9; k^:=9;
    eins(i^,j^,k^);
    writeln(i^,j^,k^);
    eins(k^,j^,j^);
    writeln(i^,j^,k^);
    zwei(i,j,k);
    writeln(i^,j^,k^);
    zwei(k,k,k);
    writeln(i^,j^,k^);
    zwei(i,k,j);
    writeln(i^,j^,k^);
    vier(i^,j^);
    writeln(i^,j^,k^);
    vier(j^,k^);
    writeln(i^,j^,k^);
end.

```

(10.3) [6 Punkte]

Schreiben und kommentieren Sie ein Pascalprogramm, mit dem Sie die Datei **daten.txt** einlesen, die Sie im Stud.IP finden. Dann soll jeder Großbuchstabe durch den entsprechenden Kleinbuchstaben ersetzt und jeder Vokal durch einen **.** ersetzt werden. Schließlich ist das Ergebnis in die Datei **neudaten.txt** zu schreiben.

Hinweis: Verwenden Sie anstelle der Befehle **REWRITE(f, 'Name der Datei')**; bzw. **RESET(f, 'Name der Datei')**; die Befehle **ASSIGN(f, 'Name der Datei')**; **REWRITE(f)**; bzw. **ASSIGN(f, 'Name der Datei')**; **RESET(f)**;

(10.4) [2 Zusatzpunkte]

Mit acht ganzen und vier halben Streichhölzern soll ein Quadrat gebildet werden, welches aus vier gleich großen Rechtecken und einem halb so großen Quadrat zusammengesetzt ist.

Gesamtpunktzahl: 22

Abgabetermin: 21.01.2010