

## Musterlösung

### Aufgabe 1

Gegeben sei das folgende Programm:

```
program WasPassiert(input,output);  
{ Das Programm multipliziert zwei natürliche Zahlen a und b  
}  
type  
tNatZahl=1..maxint;  
var  
a:tNatZahl;  
b:tNatZahl;  
c:integer;  
begin  
  writeln('Geben Sie zwei natürliche Zahlen ein:');  
  readln(a);  
  readln(b);  
  c:=0;  
  while (b > 1) do  
    begin  
      if (b mod 2 = 1) then  
        begin  
          c:=c+a;  
          b:=b-1  
        end;  
        a:=2*a;  
        b:=b div 2  
      end;  
    writeln('Ergebnis: ',a+c)  
  end.
```

Überlegen Sie sich was das Programm leistet und wie es dabei vorgeht.

- a) Was gibt das Programm für die Eingaben  $a=3$  und  $b=4$  aus? 'Ergebnis: 12'  
Was gibt das Programm für die Eingaben  $a=6$  und  $b=7$  aus? 'Ergebnis: 42'
- b) Ergänzen Sie im Programm einen erklärenden Kommentar an der grau eingefärbten Stelle und schreiben Sie eine passende Problemspezifikation:

Eingabe: a und  $b \in \mathbb{N}$

Ausgabe:  $c \in \mathbb{N}$

Nachbedingung:  $c=a*b$

## Musterlösung

### Aufgabe 2

```
program Zinsen(input,output);  
{Das Programm berechnet das jährlich angesparte Geld aus  
Anlagebetrag, Zinsen und Laufzeit.}  
  
  var  
  Betrag:real;  
  Laufzeit:integer;  
  Zinsen:integer;  
  i:integer;  
  
begin  
  writeln('Geben Sie Betrag, Laufzeit und Zinsen ein: ');  
  readln(Betrag);  
  readln(Laufzeit);  
  readln(Zinsen);  
  for i:=1 to Laufzeit do  
  begin  
    Betrag:=Betrag*(1+Zinsen/100);  
    writeln('Nach ', i, ' Jahren ', Betrag:2:2);  
  end  
end.
```

## Musterlösung

### Aufgabe 3

**function** vgl(inFeldA:tFeld;inFeldB:tFeld):boolean;  
{Die Funktion überprüft ob alle Werte des Feldes inFeldA auch im  
Feld inFeldB vorkommen.}

**var**

gefunden:boolean;  
alleGefunden:boolean;  
i:tIndex;  
j:tIndex;

**begin**

alleGefunden:=true;  
for i:=1 to FELDGROESSE do  
begin  
gefunden:=false;  
for j:=1 to FELDGROESSE do  
if (inFeldA[i]=inFeldB[j]) then  
gefunden:=true;  
if (gefunden=false) then  
alleGefunden:=false  
end;  
vgl:=alleGefunden  
end;

## Musterlösung

### Aufgabe 4

```
procedure ListeUmdrehen(var ioListe:tRefListe);  
{Die Prozedur dreht die Elemente der Liste ioListe.}  
  
  var  
    Zeiger : tRefListe;  
    ZeigerAlt : tRefListe;  
  
begin  
  ZeigerAlt:=nil;  
  while ioListe<>nil do  
    begin  
      Zeiger:=ioListe;  
      ioListe:=ioListe^.next;  
      Zeiger^.next:=ZeigerAlt;  
      ZeigerAlt:=Zeiger  
    end;  
  ioListe:=ZeigerAlt  
end;
```

## Musterlösung

### Aufgabe 5

```
function SucheBinBaum(inWert:integer;inBaum:tRefBinBaum):boolean;  
{Die Funktion entscheidet ob der Wert inWert im Baum inBaum  
enthalten ist.}
```

```
begin
```

```
  if inBaum<>nil then
```

```
    SucheBinBaum:=((inBaum^.Wert=inWert)OR
```

```
                  SucheBinBaum(inWert,inBaum^.links)OR
```

```
                  SucheBinBaum(inWert,inBaum^.rechts))
```

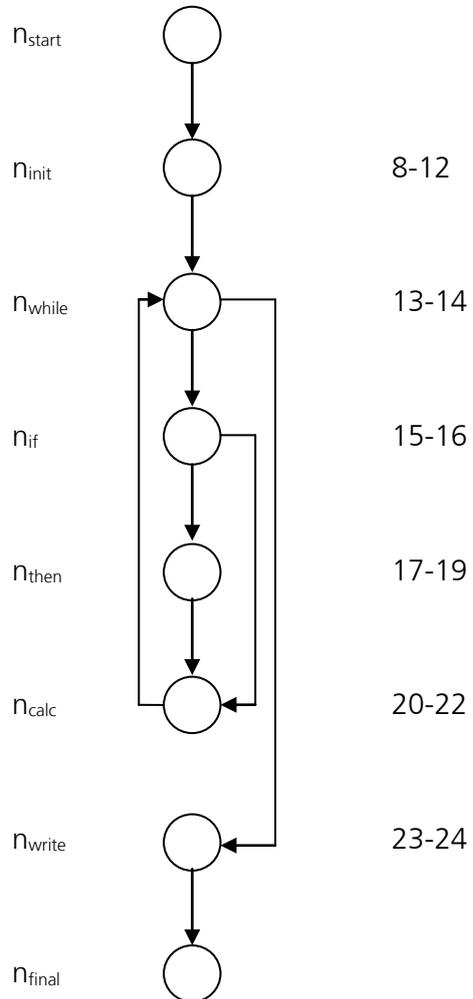
```
  else
```

```
    SucheBinBaum:=false
```

```
end;
```

Musterlösung

**Aufgabe 6**



Eingabedatum für 0 Schleifendurchläufe:

(4,1)

Eingabedaten für 1 Schleifendurchlauf:

(4,2), (4,3)

Eingabedaten für 2 Schleifendurchläufe:

(4,4), (4,5), (4,6), (4,7)