

Veranstaltung
Pr.-Nr.: 10 1023 V

Wirtschaftsinformatik für
Wirtschaftswissenschaftler

Algorithmen und ihre **Programmierung**

- Teil 2 -

Dr. Chris Bizer
WS 2007/2008

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Gliederung

- 1. Funktionen und Prozeduren**
2. Beispiele zu Funktionen und Prozeduren
- 3. Arrays**
4. Beispiele zu Arrays
5. Klausuraufgabe: Lotto 6 aus 49
- 6. Geschachtelte Arrays**
7. Klausuraufgabe: Rinderwahn

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Funktionen und Prozeduren

- Funktionen und Prozeduren dienen der
 - besseren Gliederung längerer Programme.
 - besseren Wartbarkeit von Programmen.
 - der Verknüpfung von Programmcode mit Steuerelementen, wie beispielsweise einem Button.

- **Prozedur (auch Subroutine genannt)**

- Prozeduren sind Programmfragmente, die über ihren Namen aufgerufen werden können.
- Prozeduren können ein oder mehrere Werte (Argumente) übergeben werden, die ihr Verhalten beeinflussen.

- **Syntax**

```
Private Sub [Prozedurname]([Argumentenliste])  
    [Anweisungsblock]  
End Sub
```

- **Beispiel**

```
Private Sub CommandButton2_Click()  
    MsgBox "Sie haben den zweiten Knopf gedrückt!"  
End Sub
```

- **Aufruf der Prozedur:** **Syntax:** Call [Prozedurname]([Argumentenliste])
Beispiel: Call CommandButton2_Click

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Funktionen und Prozeduren

- **Funktionen**

- Funktionen sind Programmfragmente, denen ein oder mehrere Werte (Argumente) übergeben werden und die ein Ergebnis (Rückgabewert) zurückliefern.
- Der Rückgabewert ergibt sich, indem man innerhalb der Funktion der Variablen [Funktionsname] einen Wert zuweist.

- **Syntax**

```
Private Function [Funktionsname]([Argumentenliste])  
    [Anweisungsblock]  
    [Funktionsname] = Wert  
End Function
```

- **Beispiel**

```
Private Function Quadrat(a as Integer)  
    Quadrat = a * a  
End Function
```

- **Aufruf einer Funktion:** **Syntax:** Variable = [Funktionsname]([Argumentenliste])
Beispiel: ergebnis = Quadrat(5)

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Beispiel 1: Funktionen und Prozeduren

```
Private Sub CommandButton1_Click()  
    Dim zahl As Integer, ergebnis As Integer  
    zahl = 4  
    ergebnis = Rechnen(zahl)  
    ergebnis = Rechnen(ergebnis)  
    Call Ausgabe(ergebnis)  
End Sub  
  
Private Function Rechnen(a As Integer)  
    Dim b as Integer  
    b = a/2  
    Rechnen = a + b  
End Function  
  
Private Sub Ausgabe(a As Integer)  
    MsgBox "Variablenwert: a=" & a  
End Sub
```

Ausgabe: Variablenwert a=9

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Beispiel 2: Funktionen und Prozeduren

- Die Prozedur `CommandButton1_Click()` fragt den Benutzer nach einer Zahl und rechnet anschließend, unter Verwendung der Funktionen `Quadrat()` und `HochDrei()`, $zahl^2$ und $zahl^3$ aus.

```
Private Sub CommandButton1_Click()  
    Dim zahl As Integer, zahl2 As Integer, zahl3 As Integer  
    zahl = Inputbox("Bitte geben Sie eine Zahl ein:")  
    zahl2 = Quadrat(zahl)  
    zahl3 = HochDrei(zahl)  
    Call Ausgabe(zahl2, zahl3)  
End Sub  
  
Private Function Quadrat(a As Integer)  
    Quadrat = a * a  
End Function  
  
Private Function HochDrei(a As Integer)  
    HochDrei = a * a * a  
End Function  
  
Private Sub Ausgabe(a As Integer , b As Integer)  
    MsgBox „Ergebnis: Eingabe hoch 2 = " & a & "; hoch 3 = " & b  
End Sub
```

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Arrays

Ein Array ist eine **Datenstruktur**, die aus einer Menge durchnummerierter Variablen besteht.

- Anders gesagt: Ein Array ist eine Liste, deren Elemente sich über den Listennamen und die Nummer des Elements ansprechen lassen.
- Ein Array definieren:
 - Syntax `Dim [Arrayname]([MinNummer] to [MaxNummer]) as [Datentyp]`
 - Beispiel `Dim Messung(1 to 5) as Integer`
- Auf ein Array-Element zugreifen
 - Syntax `[Arrayname]([Nummer des Elements])`
 - Beispiel

```
Dim Messung(1 to 5) as Integer
Messung(1) = 5
Messung(2) = 3
MsgBox "Zweite Messung" & Messung(2)
```

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Beispiel 1: Arrays

```
Dim zahlen(1 To 4) As Integer
zahlen(1) = 5
zahlen(2) = 7
zahlen(3) = 1
zahlen(4) = 3
While zahlen(2) > zahlen(1)
    zahlen(2) = zahlen(2) - 1
    zahlen(3) = zahlen(2)
Wend
zahlen(4) = zahlen(3)
```

Tischtest

zahlen(1)	zahlen(2)	zahlen(3)	zahlen(4)
0	0	0	0
5			
	7		
		1	
			3
	6		
		6	
	5		
		5	
			5

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Beispiel 2: Summieren von vier Zahlen

- Die Prozedur `Summieren()` fragt den Nutzer nach vier Zahlen und errechnet die Summe dieser Zahlen.

```
Private Sub Summieren()  
    Dim eingaben(1 To 4) As Single  
    Dim a, summe As Integer  
    For a = 1 To 4  
        eingaben(a) = InputBox("Bitte geben Sie die " & a & ". Zahl ein:")  
    Next a  
    summe = 0  
    For a = 1 To 4  
        summe = summe + eingaben(a)  
    Next a  
    MsgBox "Die Summe der Eingaben ist " & summe  
End Sub
```

- **Wichtig:** Immer darauf achten, dass der Schleifenzähler innerhalb der Grenzen des Arrays bleibt, da es sonst einen "Index außerhalb des gültigen Bereichs"-Fehler gibt!

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Beispiel 3: Telefonverzeichnis

- Ein Unternehmen hat ein Telefonverzeichnis erstellt. Es handelt sich dabei um eine Liste mit 1000 Einträgen, bestehend aus Namen und Telefonnummern.
- Der folgende Algorithmus soll für einen gegebenen Namen, der in der Variablen `suchname` gespeichert ist, die entsprechende Telefonnummer aus dem Telefonverzeichnis suchen und ausgeben.
- Dabei sind die Telefonnummern in dem Array `telefonnummern(1 To 1000)` as `Long` und die Namen in dem Array `namen(1 To 1000)` as `String` gespeichert.

```
Private Sub Suchen(suchname, telefonnummern, namen)  
    Dim index As Integer  
    For index = 1 to 1000  
        If suchname = namen(index) Then  
            MsgBox telefonnummern(index)  
        End If  
    Next index  
End Sub
```

Beispieldaten

x	namen(x)	telefonnummern(x)
1	Müller	8254767
2	Meier	6948314
3	Schulz	5476776
4	Schmidt	3287534

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Beispiel 4: EAN 8

- Die 8-stellige Artikel-Nummer (EAN) besteht aus 8 Dezimalziffern. Die Nummerierung der Stellen erfolgt von links nach rechts. Ziffer 8 ist eine Prüfziffer, die aus den anderen Ziffern errechnet wird und zur Kontrolle der korrekten Datenerfassung dient.
- Beispiel:

EAN	4	0	1	2	3	4	5	5
Gewichtung	3	1	3	1	3	1	3	
Produkte	12	0	3	2	9	4	15	



- Produktsumme = $12 + 0 + 3 + 2 + 9 + 4 + 15 = 45$
- Rest der ganzzahligen Division: Produktsumme MOD 10 = 5
- Prüfziffer = Rest = 5.
- EAN-Prüfung erfolgreich!

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Beispiel 4: EAN 8

```

...
'Berechne die Prüfsumme an den geraden Positionen
Pruefsumme = 0
i = 2
While i < 8
    Pruefsumme = Pruefsumme + ean(i)
    i = i + 2
Wend
'Berechne die Prüfsumme an den ungeraden Positionen
i = 1
While i < 8
    Pruefsumme = Pruefsumme + ean(i) * 3
    i = i + 2
Wend
'Berechne die Prüfziffer
Pruefziffer = Pruefsumme Mod 10
'Ausgabe des Prüfergebnisses
If Pruefziffer = ean(8) Then
    MsgBox "EAN-Prüfung erfolgreich!"
Else
    MsgBox "EAN-Prüfung nicht erfolgreich!"
End If
...

```

Beispieldaten

x	ean(x)	Faktoren
1	4	3
2	0	1
3	1	3
4	2	1
5	3	3
6	4	1
7	5	3
8	5	

- Der nebenstehende Code ist ein Teil der Prozedur: **PruefeEan8()**
- Ergänzen Sie ihn so, dass die EAN Prüfung korrekt durchgeführt wird.

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Klausuraufgabe: 6 aus 49 - Gewinnhäufigkeit

- Die Lottogesellschaft (6 aus 49 – Lotto) will die Häufigkeiten der Gewinnzahlen ermitteln, die in den letzten 52 Ziehungen gezogen wurden.
- Dazu wird das Ergebnis der 52 Gewinnzahlenziehungen in dem Array `ziehungen` mit 312 Elementen gespeichert.
- Die Funktion `Gewinnhäufigkeiten(ziehungen)` soll die Häufigkeiten der Gewinnzahlen in dem Array `anzahl` mit 49 Elementen speichern.
- Abschließend soll diejenige bzw. diejenigen Gewinnzahlen bestimmt werden, die am häufigsten gezogen wurden.
- Bitte beachten Sie, dass auch gleichzeitig mehrere Gewinnzahlen am häufigsten gezogen worden sein können.

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Klausuraufgabe: 6 aus 49 - Gewinnhäufigkeit

```
Private Sub Gewinnzahlenhäufigkeiten (ziehungen)
Dim ziehungen(1 To 312) As Integer, anzahl(1 To 49)
As Integer, x As Integer, y As Integer, gewinnzahl
As Integer, maxAnzahl As Integer
x = 1
While AAA
    Anzahl(x) = 0
    x = x + 1
Wend
x = 1
While BBB
    gewinnzahl = ziehungen(x)
    anzahl(gewinnzahl) = CCC
    x = x + 1
Wend
y = 1
maxAnzahl = 0
While y<50
    If DDD maxAnzahl Then
        maxAnzahl = anzahl(y)
    End If
    y = y + 1
Wend
y = 0
While FFF
    y = y + 1
    If EEE Then
        MsgBox "Gewinnzahl (mit Anzahl):" & y & anzahl(y)
    Endif
Wend
End Sub
```

Aufgabe 7.1 (2P): Der Platzhalter AAA kann durch folgende(n) Term(e) ersetzt werden:

- a) $x < 50$
- b) $x < 49$
- c) $x \leq 49$
- d) $x < 51$

Aufgabe 7.2 (2P): Der Platzhalter BBB kann durch folgende(n) Term(e) ersetzt werden:

- a) $x = 312$
- b) $x \leq 312$
- c) $\text{anzahl}(x) \leq 49$
- d) $\text{ziehungen}(x) \leq 49$

Klausuraufgabe: 6 aus 49 - Gewinnhäufigkeit

```

Private Sub Gewinnzahlenhäufigkeiten (ziehungen)
Dim ziehungen(1 To 312) As Integer, anzahl(1 To 49)
As Integer, x As Integer, y As Integer, gewinnzahl
As Integer, maxAnzahl as Integer
x = 1
While AAA
    Anzahl(x) = 0
    x = x + 1
Wend
x = 1
While BBB
    gewinnzahl = ziehungen(x)
    anzahl(gewinnzahl) = CCC
    x = x + 1
Wend
y = 1
maxAnzahl = 0
While y<50
    If DDD maxAnzahl Then
        maxAnzahl = anzahl(y)
    End If
    y = y + 1
Wend
y = 0
While FFF
    y = y + 1
    If EEE Then
        MsgBox "Gewinnzahl (mit Anzahl):" & y & anzahl(y)
    Endif
Wend
End Sub

```

Aufgabe 7.3 (3P): Der Platzhalter CCC kann durch folgende(n) Term(e) ersetzt werden:

- a) anzahl(gewinnzahl) + anzahl(x)
- b) 1 + anzahl(gewinnzahl)
- c) 1 + anzahl(x)
- d) anzahl(gewinnzahl) + ziehungen(x)

Aufgabe 7.4 (3P): Der Platzhalter DDD kann durch folgende(n) Term(e) ersetzt werden:

- a) anzahl(y) >
- b) ziehungen(y) >
- c) anzahl(maxAnzahl) >=
- d) ziehungen(maxAnzahl) >=

Aufgabe 7.2 und 7.3

ziehungen(1)	ziehungen(2)	ziehungen(3)	ziehungen(4)	...
2	34	4	2	...

```

....
x = 1
While x <= 312
    gewinnzahl = ziehungen(x)
    anzahl(gewinnzahl) = 1 + anzahl(gewinnzahl)
    x = x + 1
Wend
....

```

anzahl(1)	anzahl(2)	anzahl(3)	anzahl(4)	...
0	0	0	0	...
0	1	0	0	...
0	1	0	1	
0	2	0	1	

Aufgabe 7.4

anzahl(1)	anzahl(2)	anzahl(3)	anzahl(4)	...
5	12	8	30	...

```

y = 1
maxAnzahl = 0
While y < 50
  If anzahl(y) > maxAnzahl Then
    maxAnzahl = anzahl(y)
  End If
  y = y + 1
Wend
  
```

maxAnzahl
0
5
12
12
30
...

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Klausuraufgabe: 6 aus 49 - Gewinnhäufigkeit

```

Private Sub Gewinnzahlenhäufigkeiten (ziehungen)
Dim ziehungen(1 To 312) As Integer, anzahl(1 To 49)
As Integer, x As Integer, y As Integer, gewinnzahl
As Integer, maxAnzahl as Integer
x = 1
While AAA
  Anzahl(x) = 0
  x = x + 1
Wend
x = 1
While BBB
  gewinnzahl = ziehungen(x)
  anzahl(gewinnzahl) = CCC
  x = x + 1
Wend
y = 1
maxAnzahl = 0
While y<50
  If DDD maxAnzahl Then
    maxAnzahl = anzahl(y)
  End If
  y = y + 1
Wend
y = 0
While FFF
  y = y + 1
  If EEE Then
    MsgBox "Gewinnzahl (mit Anzahl):" & y & anzahl(y)
  Endif
Wend
End Sub
  
```

Aufgabe 7.5 (3P): Der Platzhalter EEE kann durch folgende(n) Term(e) ersetzt werden:

- a) $\text{anzahl}(y) \geq \text{maxAnzahl}$
- b) $\text{ziehungen}(y) \geq \text{maxAnzahl}$
- c) $\text{anzahl}(\text{maxAnzahl}) = \text{maxAnzahl}$
- d) $\text{ziehungen}(\text{maxAnzahl}) = \text{maxAnzahl}$

Aufgabe 7.6 (2P): Der Platzhalter FFF kann durch folgende(n) Term(e) ersetzt werden:

- a) $y \leq 50$
- b) $y \leq 49$
- c) $y < 49$
- d) $y > 51$

Lösung: 6 aus 49 - Gewinnhäufigkeit

Die richtigen Antworten lauten:

7.1 a) $x < 50$

c) $x \leq 49$

7.2 b) $x \leq 312$

7.3 b) $1 + \text{anzahl}(\text{gewinnzahl})$

7.4 a) $\text{anzahl}(y) >$

7.5 a) $\text{anzahl}(y) \geq \text{maxAnzahl}$

7.6 c) $y < 49$

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Geschachtelte Arrays

■ Mehrere Array lassen sich auch ineinander schachteln.

■ Beispiel: Array a() und Array b()

x	a(x)
1	3
2	2
3	5
4	2
5	8
6	7

x	b(x)
1	2
2	3
3	6
4	3
5	7
6	5

■ Geschachtelter Arrayausdruck: **MsgBox a(b(3))**

1. $a(b(3))$ und $b(3) = 6$, daraus folgt:
2. $a(6)$, daraus folgt:
3. Der Variablenwert ist 7.

■ $a(b(3)) = 7$

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Beispiel: Geschachtelte Arrays

Array a()

x	a(x)
1	3
2	2
3	5
4	2
5	8
6	7

und

Array b()

x	b(x)
1	2
2	3
3	6
4	3
5	7
6	5

und

Array c()

x	c(x)
1	4
2	8
3	2
4	4
5	1
6	0

■ Inhalt der Array Elemente:

1. $a(3) = 5$
2. $a(c(1)) = 2$
3. $b(c(3)) = 3$
4. $c(b(1)) = 8$
5. $a(b(c(3))) = 5$
6. $b(c(2))$ Fehler: Index außerhalb des gültigen Bereichs

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

Klausuraufgabe: Rinderwahn

- Deutschland im Jahr 2002. Der Rinderwahn greift weiter um sich. Der Bund beschließt alle noch verbleibenden 1.800.000 Rinder auf BSE zu testen. Die Rinder befinden sich in insgesamt 4500 Herden, die von Landwirten oder großen Rindermastbetrieben bewirtschaftet werden. Es soll ein computergestütztes Informationssystem erstellt werden, das einige Kennzahlen ermittelt.
- Der folgende Algorithmus soll eine Herdennummer mit der größten Anzahl BSE-infizierter Tiere ermitteln.
- Die Ergebnisse der BSE-Tests sind im Array BSE gespeichert. Befindet sich eine Null in dem k-ten Element, dann konnte bei dem k-ten Rind kein BSE-Erreger festgestellt werden. Wird eine BSE-Erkrankung diagnostiziert, dann wird an der k-ten Position des Arrays BSE die Herdennummer des Rindviehs gespeichert. Jede Herdennummer gehört zur Menge $\{1, 2, \dots, 4500\}$.

BSE	0	0	0	0	12	0	0	1100	0	0	0	0	0	0	3345	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	...

Beispiel: Das 5. Rind ist an BSE erkrankt und gehört zur Herde mit der Nummer 12.

Freie Universität Berlin – Bizer: Wirtschaftsinformatik – WS07/08 (Version vom 28.1.08)

BSE	0	0	0	0	12	0	0	1100	0	0	0	0	0	0	3345	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	...

```

Private Sub Herde(BSE)
    Dim BSE(1 To 1800000) As Integer,
        Rinderherde(1 To 4500) As Integer, k As Long,
        Herdennummer As Integer, maxAnzahl As Integer
    k = 1
    While AAA
        Rinderherde(k) = 0
        k = k + 1
    Wend
    k = 0
    While DDD
        k = k + 1
        If BBB Then
            CCC
        End If
    Wend
    k = 0
    maxAnzahl = 0
    While k < 4500
        k = k + 1
        If EEE Then
            maxAnzahl = Rinderherde(k)
            Herdennummer = k
        End If
    Wend
    MsgBox " Eine Herdennummer mit der größten Anzahl
    BSE-Infizierungen: " & Herdennummer
End Sub

```

Aufgabe 8.1 (4P): Welche Platzhalter für AAA sind richtig?

- a) $4500 > k$
- b) $4500 < k$
- c) $4500 = k$
- d) $k < 4501$

Aufgabe 8.2 (4P): Welche Platzhalter für BBB sind richtig? Die IF-Bedingung soll wahr sein, wenn eine BSE-Infizierung vorliegt.

- a) $0 < \text{BSE}(k)$
- b) $0 < \text{rinderherde}(\text{BSE}(k))$
- c) $0 < \text{BSE}(\text{rinderherde}(k))$
- d) $\text{BSE}(k) = \text{rinderherde}(k)$

BSE	0	0	0	0	12	0	0	1100	0	0	0	0	0	0	3345	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	...

```

Private Sub Herde(BSE)
    Dim BSE(1 To 1800000) As Integer,
        Rinderherde(1 To 4500) As Integer, k As Long,
        Herdennummer As Integer, maxAnzahl As Integer
    k = 1
    While AAA
        Rinderherde(k) = 0
        k = k + 1
    Wend
    k = 0
    While DDD
        k = k + 1
        If BBB Then
            CCC
        End If
    Wend
    k = 0
    maxAnzahl = 0
    While k < 4500
        k = k + 1
        If EEE Then
            maxAnzahl = Rinderherde(k)
            Herdennummer = k
        End If
    Wend
    MsgBox " Eine Herdennummer mit der größten Anzahl
    BSE-Infizierungen: " & Herdennummer
End Sub

```

Aufgabe 8.3 (4P): Welche Platzhalter für CCC sind richtig? Berechnen Sie die Anzahl der BSE-Infizierungen je Herde.

- a) $\text{rinderherde}(\text{BSE}(k)) = \text{rinderherde}(\text{BSE}(k)) + 1$
- b) $\text{rinderherde}(k) = \text{rinderherde}(k) + 1$
- c) $\text{BSE}(k) = \text{BSE}(k) + 1$
- d) $\text{BSE}(k) = \text{rinderherde}(\text{BSE}(k)) + 1$
- e) Keine der Antworten von a) bis d) ist richtig.

BSE	0	0	0	0	12	0	0	1100	0	0	0	0	0	0	3345	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	...

```

Private Sub Herde(BSE)
  Dim BSE(1 To 1800000) As Integer,
  Rinderherde(1 To 4500) As Integer, k As Long,
  Herdennummer As Integer, maxAnzahl As Integer
  k = 1
  While AAA
    Rinderherde(k) = 0
    k = k + 1
  Wend
  k = 0
  While DDD
    k = k + 1
    If BBB Then
      CCC
    End If
  Wend
  k = 0
  maxAnzahl = 0
  While k < 4500
    k = k + 1
    If EEE Then
      maxAnzahl = Rinderherde(k)
      Herdennummer = k
    End If
  Wend
  MsgBox " Eine Herdennummer mit der größten Anzahl
  BSE-Infizierungen: " & Herdennummer
End Sub

```

Aufgabe 8.4 (4P): Welche Platzhalter für DDD sind richtig?

- a) k < 1800000
- b) k < 4500
- c) k = 1800001
- d) k > 1800000
- e) Keine der Antworten

Aufgabe 8.5 (4P): Welche Platzhalter für EEE sind richtig? Ermitteln Sie eine Herdennummer mit der größten Anzahl von BSE-Infizierungen.

- a) rinderherde(k) > maxAnzahl
- b) k > maxAnzahl
- c) BSE(k) > maxAnzahl
- d) rinderherde(k) > BSE(k)

Aufgabe 8.2 bis 8.4

BSE(1)	BSE(2)	BSE(3)	BSE(4)	...
2	0	4	2	...

```

....
While k < 1800000
  k = k + 1
  If 0 < BSE(k) Then
    rinderherde(BSE(k)) = rinderherde(BSE(k)) + 1
  End If
Wend
....

```

rinderherde(1)	rinderherde(2)	rinderherde(3)	rinderherde(4)	...
0	0	0	0	...
0	1	0	0	...
0	1	0	1	
0	2	0	1	

Lösung: Rinderwahn

Die richtigen Antworten lauten:

8.1 d) $k < 4501$

8.2 a) $0 < \text{BSE}(k)$

8.3 a) $\text{rinderherde}(\text{BSE}(k)) = \text{rinderherde}(\text{BSE}(k)) + 1$

8.4 a) $k < 1800000$

8.5 a) $\text{rinderherde}(k) > \text{maxAnzahl}$