

Veranstaltung  
Pr.-Nr.: 101023

# Algorithmen und ihre Programmierung -Teil 2-

Veronika Waue  
WS 07/08

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Wiederholung: Algorithmen

### ■ Variablen und Datentypen

- Variablen haben einen Namen, einen Datentypen und einen Wert
- Beispiel: Dim a as integer

a = 77

### ■ Operatoren

- Zuweisungsoperator (=)
- Mathematische Operatoren (\*, /, +, -, ^, \, mod)
- Relationale Operatoren (<, >, >=, <=, =, <>)
- Logische Operatoren (AND, OR, XOR)

### ■ Ein- und Ausgabebefehle

- Beispiel: x = Inputbox("Bitte geben Sie eine Zahl ein:")  
MsgBox "Ihr Guthaben beträgt " & guthaben
- Beispiel: x = Worksheets("Tabelle1").Range("D2").Value  
Worksheets("Tabelle1").Range("D2").Value = x

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Inhalte

- Variablen, Datentypen und Wertzuweisungen
- Ein- und Ausgabebefehle
- Operatoren
- Entscheidungsstrukturen
- Schleifenstrukturen
- Prozeduren und Funktionen

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Entscheidungsstrukturen

- Bestimmte Anweisungen werden nur ausgeführt, wenn eine Bedingung erfüllt ist.

```
x = 4
If x > 3 Then
    Anweisungsblock
End If
```

- Andere Anweisungen werden nur ausgeführt, wenn eine Bedingung Falsch ist

```
x = 1
If x > 3 Then
    Anweisungsblock
Else
    Anweisungsblock2
End If
```

- Die Bedingung darf nur Wahr oder Falsch sein

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Die Bedingung

### ■ Vergleichsoperatoren

Operator	Bedeutung	Operator	Bedeutung
=	Ist gleich	>=	Größer oder gleich; mindestens
>	Größer als	<=	Kleiner oder gleich; höchstens
<	Kleiner als	<>	Ungleich

### ■ logische Operatoren

- mit logischen Operatoren lassen sich zusammengesetzte Bedingungen formulieren.
- Logisches Und:
  - If [Bedingung1] **And** [Bedingung2] Then ...
  - Beide Bedingungen müssen erfüllt sein.
- Logischer Oder:
  - If [Bedingung1] **Or** [Bedingung2] Then ...
  - Eine der Bedingungen oder beide Bedingungen müssen erfüllt sein.
- Exklusives Oder:
  - If [Bedingung1] **Xor** [Bedingung2] Then ...
  - Eine der Bedingungen, aber nicht beide Bedingungen müssen erfüllt sein.

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Übung 2

```
Dim Zahl(1 To 10) As Integer
Dim x, a As Integer
Private Sub cmdIfAbfr_Click()
x = 3
a = 4
If x > a Then
    x = x - 1
    Zahl(x) = 3
    a = Zahl(x)
Else
    Zahl(x) = 5
    x = x - 2
    Zahl(a - 1) = 2
End If
End Sub
```

■ Es soll davon ausgegangen werden, dass im Array Zahl zu Beginn an allen Stellen Nullen stehen. Welchen Wert haben die Variablen nach der If-Abfrage, wenn:

1. a = 4 und x = 3
2. a = 2 und x = 5?

■ Lösung

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Elseif

- Mit Elseif lassen sich mehrere Bedingungen nacheinander prüfen.
- Sobald eine Bedingung wahr ist werden die übrigen nicht mehr getestet

```
If x < a Then
    Anweisungsblock1
ElseIf x > a Then
    Anweisungsblock2
Else
    Anweisungsblock3
End If
```

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Übung 3

```
Dim c(1 To 2) As Integer
Dim a, b As Integer
a = 2
b = 14
c(1) = 3
c(2) = 5
If b / 5 < a Then
    c(a) = a * b
ElseIf b / 2 < 3 * a Then
    c(a) = b + 1
Else
    a = a - 1
    c(a + 1) = b
End If
If c(2) > a And c(2) <= b Then
    c(2) = c(a) + b
    b = 0
End If
If c(a) = a Or b > a Then
    b = c(a - 1) + a
End If
```

**Welches Ergebnis liefert das Programm?**

**Welches Ergebnis liefert das Programm, wenn b= 14?**

**1.Tischtest**

a	b	c(1)	c(2)
2	9	3	5

**2.Tischtest**

a	b	c(1)	c(2)
2	14	3	5

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Zusammenfassung If- Abfragen

<b>1. Anweisungsblock1 wird nur ausgeführt, wenn die Bedingung wahr ist.</b>	<pre>If Bedingung1 Then     Anweisungsblock1 End If</pre>
<b>2. Es wird entweder Anweisungsblock1 oder Anweisungsblock4 ausgeführt abhängig davon, ob die Bedingung wahr oder falsch ist.</b>	<pre>If Bedingung1 Then     Anweisungsblock1 Else     Anweisungsblock4 End If</pre>
<b>3. Es wird der Anweisungsblock ausgeführt, dessen Bedingung wahr ist, ist keine der Bedingungen wahr wird auch kein Anweisungsblock ausgeführt</b>	<pre>If Bedingung1 Then     Anweisungsblock1 ElseIf Bedingung2 Then     Anweisungsblock2 ElseIf Bedingung3 Then     Anweisungsblock3 End If</pre>
<b>4. Es wird der Anweisungsblock ausgeführt, dessen Bedingung wahr ist, ist keine der Bedingungen wahr wird Anweisungsblock3 ausgeführt.</b>	<pre>If Bedingung1 Then     Anweisungsblock1 ElseIf Bedingung2 Then     Anweisungsblock2 Else     Anweisungsblock3 End If</pre>

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Inhalte

- Variablen, Datentypen und Wertzuweisungen
- Ein- und Ausgabebefehle
- Operatoren
- Entscheidungsstrukturen
- Schleifenstrukturen
- Prozeduren und Funktionen

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Schleifen

- Mittels Schleifen können Anweisungsblöcke mehrfach ausgeführt werden.
- Wir behandeln:
  - For... Next Schleifen, die eine bestimmte Anzahl von Durchläufen angibt

```
For a = 1 To 3
    Anweisungsblock
Next
```

- While [Bedingung]... Wend Schleifen, durch die ein Anweisungsblock solange ausgeführt wird, wie eine Bedingung wahr

```
a = 1
While a < 10
    Anweisungsblock
    a = a + 1
Wend
```

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## For .. Next Schleife

- Schleife mit einer feststehenden Anzahl von Durchläufen

```
Dim a, i As Integer
a = 2
For i = 1 To 3
    If a <= 4 Then
        a = a + 2 * i
    End If
Next
```

Implizite Wertzuweisung  
an die Zählvariable i

Anweisungsblock

Hochzählen der  
Zählvariablen i

Tischtest

a	i
2	1
4	2
8	3
	4

- Bei Erreichen des Wortes NEXT wird i hochgezählt und es wird nach oben zum FOR gesprungen und geprüft, ob das Ende der Schleife erreicht ist:
  - Ist das Ende erreicht ( $i > 3$ ) wird die Schleife verlassen und die Anweisungen hinter dem NEXT werden ausgeführt
  - Ist das Ende noch nicht erreicht ( $i \leq 3$ ) werden die Anweisungen in der Schleife erneut ausgeführt

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Übung 4

```
Dim Tag As String
Dim i, a, b, Anz As Integer
Tag = "heute"
For i = 1 To 2
    If Tag = "heute" Then
        For a = 1 To 3
            b = a * i + b
        Next
        Tag = "morgen"
    End If
    For a = 1 To 2
        Anz = Anz + a + b
    Next
Next
```

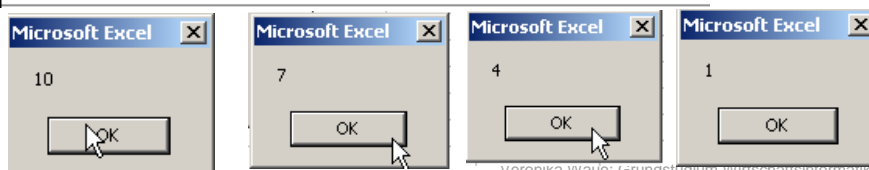
1. Wie oft wird b ein neuer Wert zugewiesen?
2. Wie oft wird Anz ein neuer Wert zugewiesen?
3. Darf für mehrer Schleifen die gleiche Zählvariable genutzt werden? (hier: a)
4. Hätte man in diesem Beispiel für alle drei Schleifen die gleiche Zählvariable nutzen können?

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## For-Schleife mit Schritten

- Normalerweise wird die Zählvariable bei einer For-Schleife, um immer genau eins hochgezählt.
- Ist eine andere Schrittlänge gewünscht, kann diese mittels dem Wort **STEP** angegeben werden.

```
Private Sub cmdStep_Click()
    Dim a, b As Integer
    For a = 10 To 1 Step -3
        MsgBox a
    Next
End Sub
```



Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## While .. Wend Schleifen

- Bei der While .. Wend Schleife ist die Anzahl der Durchläufe von einer Bedingung abhängig. Die Schleife wird so lange durchlaufen, wie die Bedingung wahr ist.

```
Dim a, b As Integer
a = 0
b = 1
While a < 5
    a = b * 2
    b = a
Wend
```

Tischtest

a	b
0	1
2	2
4	4
8	8

- Bei der While... Wend Schleife muss es immer eine **Zahlvariable** geben, die
  1. Intitialisiert wird
  2. Teil der Bedingung ist
  3. Innerhalb der Schleife verändert wird
- Beim Erreichen des Wortes WEND wird zum WHILE gesprungen und die Bedingung geprüft
  - Ist die Bedingung wahr, werden die Anweisungen in der Schleife erneut ausgeführt
  - Ist die Bedingung falsch werden die Anweisungen, die dem WEND folgen ausgeführt
- Die Schleife wird **keinmal oder mehrmals** durchlaufen.

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Beispiel *Summe*

- Algorithmus zur Berechnung von

$$\sum_{i=1}^n i = 1 + 2 + 3 + \dots + (n-1) + n$$

- Was soll gemacht werden?
  1. n wird von dem Benutzer eingegeben
  2. Wiederhole für jede Zahl i bis i > n:
  3. Addiere i zu einer Zwischensumme hinzu
  4. Am Ende soll die Ergebnissumme ausgegeben werden

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08



## Beispiel *Summe* mit While-Schleife

```
Dim n, i, summe As Integer
summe = 0
n = 
i = 1
While 
    i = i + 1
Wend
MsgBox "Das Ergebnis ist: " & summe
```

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## ...mit For-Schleife

```
Dim n, i, summe As Integer
summe = 0
n = 
For 
Next
MsgBox "Das Ergebnis ist: " & summe
```

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Beispiel Fakultät $n!$

- Algorithmus zur Berechnung der Fakultät einer Zahl  $n$ :

$$n! = n * (n - 1) * \dots * 3 * 2 * 1$$

- Warum liefert folgender Algorithmus nicht das gewünschte Resultat?

```
Dim n As Integer
Dim fak As Integer, i As Integer
n = InputBox("Geben Sie einen Wert für n ein")
fak = 1
For i = 1 To n
    fak = fak * n
Next
MsgBox "Das Ergebnis ist " & fak
```

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Schleifen und Arrays

- Durch Schleifen wird die Handhabung von Arrays vereinfacht.
- Die einzelnen Stellen eines Arrays werden mittels einer Variable in einer Schleife nach und nach angesprochen.

■ Bisher:

```
Liste(1) = b + 1
Liste(2) = b + 2
Liste(3) = b + 3
...
Liste(49) = b + 49
Liste(50) = b + 50
```

■ Mit Schleife:

```
For a = 1 To 50
    Liste(a) = b + a
Next
```

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Schleifen und Arrays II

- Immer darauf achten, dass der Schleifenzähler innerhalb der Grenzen des Arrays bleibt, da es sonst einen "Index außerhalb des gültigen Bereichs"- Fehler gibt!
- Um eine Schleife zu überprüfen, sollte der kleinstmögliche und der größtmögliche Wert den der Index annehmen kann bzw. soll geprüft werden.

```
Dim Liste(1 To 50) As Integer
Dim i As Integer
For i = 1 To 50
    Liste(i) = 1
Next
```

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Übung zur While Schleife

- Welche(r) der folgenden Algorithmen ist/sind richtig, wenn an jeder Position des Arrays *zahl()* das Wort Hallo stehen soll?

```
Dim i As Integer
Dim zahl(1 To 5) As String
```

```
i = 0
While i < 5
    i = i + 1
    zahl(i) = "Hallo"
Wend
```

```
i = 1
While i < 5
    i = i + 1
    zahl(i) = "Hallo"
Wend
```

```
i = 0
While i < 5
    i = i + 1
    zahl(i) = "Hallo"
Wend
```

```
i = 0
While i <= 5
    i = i + 1
    zahl(i) = "Hallo"
Wend
```

```
i = 0
While i <= 5
    zahl(i) = "Hallo"
    i = i + 1
Wend
```

```
i = 1
While i <= 5
    zahl(i) = "Hallo"
    i = i + 1
Wend
```

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08

## Übung 5

- In den Arrays gerade- und ungeradeZahl sollen jeweils alle geraden und ungeraden Zahlen zwischen 1 und 20 gespeichert werden. Vervollständigen folgenden Algorithmus

```
Private Sub cmdZahlen_Click()  
Dim a, b, c As Integer  
Dim geradeZahl(1 To 10) As Integer  
Dim ungeradeZahl(1 To 10) As Integer  
b = 0  
c = 0  
For   
    If a Mod 2 = 0 Then  
        geradeZahl(b) =   
    Else  
        ungeradeZahl(c) = a  
    End If  
Next  
For b = 1 To 10  
    MsgBox   
Next
```

- WS07/08

## Übung 5

- Ersetzen Sie in folgendem Beispiel die For schleife durch eine While...Wend-Schleife

```
Private Sub cmdZahlen_Click()  
Dim a, b, c As Integer  
Dim geradeZahl(1 To 10) As Integer  
Dim ungeradeZahl(1 To 10) As Integer  
b = 0  
c = 0  
For a = 1 To 20  
    If a Mod 2 = 0 Then  
        b = b + 1  
        geradeZahl(b) = a  
    Else  
        c = c + 1  
        ungeradeZahl(c) = a  
    End If  
Next  
For b = 1 To 10  
    MsgBox ungeradeZahl(b) & ", " & geradeZahl(b)  
Next
```

### Lösungsvorschläge:

- a) While a <= 20
- b) While a > 20
- c) While a = 20
- d) While a <> 20
- e) While a <= 1

Veronika Waue: Grundstudium Wirtschaftsinformatik – WS07/08