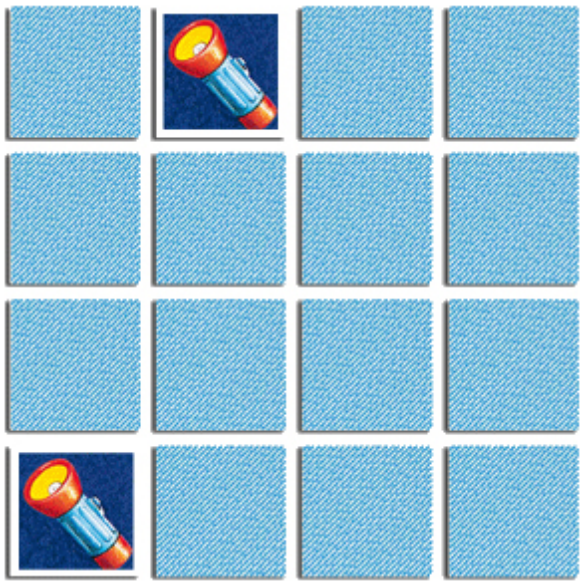
Übungseinheit: # 3

MEMORY



Entwicklungsumgebung: Excel 2007

Kursunterlagen erstell von: TECHNISCHES BÜRO MITSCH

Ing. Harald Mitsch
Josefgasse 6
A 3380 Pöchlarn

Ihr IT-Betreuer in Pöchlarn
seit 1990 - 0676/588 09 16
http://www.tb-mitsch.at

letzte Aktualisierung: 11. Jänner 2019

INHALTSVERZEICHNIS

1.) Allgemeines

Ziel dieser Übung ist es, eine Spielidee möglichst einfach in ein anspruchsvolles Spielprogramm umzusetzen. Dabei erstellen wir zuerst einen Entwurf – ohne Fehlerbehandlung und ohne alle Details wie wir sie bereits in der Übungseinheit #1 und #2 angewendet haben. Das Memory für 2 Kartenpaare (4 Karten) soll dabei voll funktionstüchtig ausprogrammiert werden und dabei in der Art gestaltet werden, damit man am Ende das Spielchen mit wenig Aufwand in ein richtiges Spielprogramm überführen kann.

2.) Spielregeln

Memory besteht aus etwa 40 bis 60 Spielkarten mit verschiedenen Motiven wobei jeweils zwei gleiche Motive ein Kartenpaar bilden. Zu Beginn werden alle Karten gemischt und verdeckt (Motiv nach unten) auf dem Spielfeld aufgelegt. Danach wird eine beliebige Karte umgedreht – Motiv ist nun sichtbar. Dann versucht man die dazugehörige Karte (mit demselben Motiv) aus dem Rest der noch verdeckten Karten zu finden und dreht auch diese Karte um. Sind die Motive der beiden Karten gleich, so bleiben die Karten aufgedeckt und der Spieler wird mit Punkten belohnt. Sind die Motive ungleich, dann werden beide Karten wieder umgedreht, der Spieler erhält keine Punkte, und versucht erneut zwei Karten für ein Kartenpaar aufzudecken. Ziel des Spieles ist es alle Paare aufzudecken und das mit möglichst wenigen Versuchen – also eine möglichst hohe Punkteanzahl zu erreichen.

3.) Überlegungen zum Spielablauf

Der Anwender soll nur auf eine Schaltfläche klicken können um das Spiel zu starten. Dabei sollen nach jedem neuen Start auch neue zufällige Motive bestimmt werden. Die Karten sollen gemischt und verdeckt auf dem Spielfeld abgelegt werden. Ein erster Klick auf eine Karte soll diese umdrehen. Der nächste Klick auf eine andere Karte dreht auch diese um. Sind die beiden Motive unterschiedlich, dann werden die beiden Karten nach ca. einer Sekunde wieder umgedreht. Ansonst soll in Abhängigkeit der vorangegangenen Fehlversuche eine Punkteanzahl berechnet werden. Gleichzeitig wollen wir dem Spieler auch anzeigen, wie viele Punkte er bereits erhalten hat. Das Spiel wird beendet, sobald alle Paare aufgedeckt worden sind. Es soll kein automatischer Neustart (wechsel auf einen höheren Level) erfolgen und auf die Hinterlegung von Sounds verzichten wir ebenfalls. Nett wäre es jedoch zusätzlich einen Highscore zu hinterlegen der auch beim Beenden des Spieles erhalten bleibt.

4.) Programmieren des Spiels

4.1.) Vorbereitungen

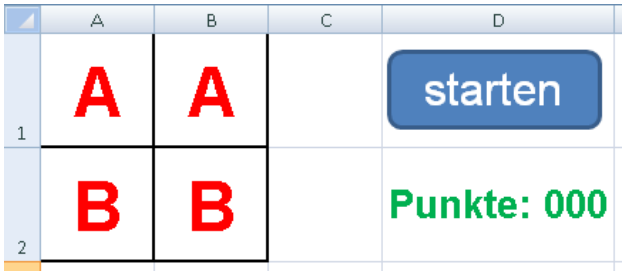
Zuerst legen wir uns einen neuen Ordner an in welchem wir alle für das Spiel benötigte Dateien ablegen. Dann kopieren wir die Datei: **Muster_Vorlage.xlsm** (die fertige Excel Datei aus der Übungseinheit #1) in diesen Ordner. Von derselben Datei legen wir uns eine weitere Kopie an und benennen sie um z.B. auf: **Memory.xlsm**. In dieser Datei werden wir am Ende das vollständige Spiel programmieren. Sollten wir dabei auf Funktionen oder andere Elemente stoßen, welche wir bei anderen Spielen ebenfalls gebrauchen könnten, dann werden wir diese zusätzlich auch in der Mustervorlage hinterlegen.

Zuerst aber erstellen wir nur einen Entwurf für unser Memory. Dazu legen wir uns eine leere Excel-Datei, z.B.: **Memory_Entwurf.xlsm**, im selben Verzeichnis an.

4.2.) Gestalten und Programmierung des Entwurfes

In der leeren Excel-Datei **Memory_Entwurf.xlsm** benennen wir das Tabellenblatt **Tabelle1** um auf **Memory** und löschen dann alle weiteren Tabellenblätter.

Gestalten Sie das Tabellenblatt **Memory** in etwa so wie rechts abgebildet. Als Schaltfläche verwenden wir wie gehabt eine beliebige Form.



Dann legen wir uns in **Visual Basic** drei Module an und benennen diese um in **Funktionen**, **Makros** und **Globales**. Fügen wir nun in jedem Modul den Programmcode für **Option Explicit** ein:

```
Option Explicit                                 'Alle Variablen muessen
                                                'explizit angegeben werden.
```

Erstellen wir nun eine Funktion für den Spielstart im Modul **Funktionen** und ein Makro im Modul **Makros** welches die Startfunktion aufruft:

```
Function SpielStarten()                         'Funktion zum Starten:
  MsgBox "Spielstart..."                        'Eine Meldung ausgeben.
End Function                                    'Funktionsende.

Public Sub Spiel_Starten()                      'Makro fuer Schaltflaeche starten
  Call SpielStarten                             'Funktion SpielStarten aufrufen.
End Sub                                         'Prozedurende.
```

Dann weisen wir der Schaltfläche (der Form bzw. dem Shape) **starten** das Makro **Spiel_Starten** zu. Klicken wir abschließend noch auf die Schaltfläche um die Funktionsweise und den Programmcode zu testen.

Jetzt erweitern wir die Funktion **SpielStarten** um die 4 Zellen (welche die 4 Karten repräsentieren sollen) zufällig mit A und B zu befüllen:

```
Function Starten()                                  'Funktion zum Starten:
Dim i                   As Integer                  'Allgemeine Zaehlvariable.
Dim iZeile              As Integer                  'Aktuelle Zeilennummer.
Dim iSpalte             As Integer                  'Aktuelle Spaltennummer.
Dim iZufall             As Integer                  'Eine Zufallszahl.
Dim stKarten            As String                   'Zeichen fuer Karteninhalte
Dim stKartenInhalt(4)   As String                   'Karteninhalte wie aufgelegt.

'MsgBox "Spielstart..."                             'Eine Meldung ausgeben.

stKarten = "AB"                                     'Karteninhalte definieren.
stKarten = stKarten & stKarten                      'Karten verdoppeln.
For i = 1 To Len(stKarten)                          'Alle Karteninhalte zum
  stKartenInhalt(i) = Mid(stKarten, i, 1)           'Mischen vorbereiten bzw.
Next i                                              'ins Kartenfeld einlesen.

Randomize (Timer)                                   'Zufallsgenerator initialisieren.
```

```
For i = 1 To 4                                          'Vorbereitetes Kartenfeld mischen
  iZufall = Int(Rnd() * 4) + 1                          'Jede Karte zumindest einmal
  stKartenInhalt(0) = stKartenInhalt(iZufall)           'vertauschen.
  stKartenInhalt(iZufall) = stKartenInhalt(i)
  stKartenInhalt(i) = stKartenInhalt(0)
Next i

Set oTabellenblatt = Sheets("Memory")                   'Akt. Tabellenblatt festlegen.
i = 1                                                   'Zaehlvariable ruecksetzen.
For iZeile = 1 To 2                                     'Der Reihe die Zellen mit den
  For iSpalte = 1 To 2                                  'Inhalt der Karten befuellen:
    oTabellenblatt.Cells(iZeile, iSpalte).Value = stKartenInhalt(i)
    i = i + 1                                           'Zaehlvariable erhoehen.
  Next iSpalte
Next iZeile
End Function                                            'Funktionsende.
```

Klicken wir wieder auf die Schaltfläche **starten** um den Programmcode neuerlich zu testen.

Nun legen wir uns im Modul **Globales** folgende Variablen und Konstanten an. Die Erklärungen dazu folgen später:

```
Public Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
                                                        'Deklaration der Prozedur:
                                                        'Sleep um den Programmcode
                                                        'eine Zeit (in Millisekunden)
                                                        'lang anzuhalten:
Global Const PROGRAMMNAME = "Memory"                    'Name des Spielprogrammes.
Global Const TABELLENBLATT = "Memory"                   'Tabellenblatt fuer Spiel.
Global Const MAX_ZEILEN = 2                             'Maximale Anzahl Zeilen.
Global Const MAX_SPALTEN = 2                            'Maximale Anzahl Spalten.
Global Const MAX_PAARE = MAX_ZEILEN * MAX_SPALTEN / 2 'Maximale Anzahl der Paare.

Global oTabellenblatt       As Object                   'Aktueller Tabellenblattname.
Global bZug1                As Boolean                  'Zug 1 wurde bereits gemacht.
Global iZeileZug1           As Integer                  'Zug 1 war in dieser Zeile.
Global iSpalteZug1          As Integer                  'Zug 1 war in dieser Spalte.
Global stInhaltZug1         As String                   'Zug 1 hatte diesen Inhalt.
Global iPunkte              As Integer                  'Aktuelle Anzahl der Punkte.
                                                        'Diverse Farbdefinitionen:
Global Const WEISS = 16777215 'Farbe weiss = RGB(255, 255, 255); Excel ColorIndex = 2
Global Const ROT = 255        'Farbe rot = RGB(255, 0, 0); Excel ColorIndex = 3
```

Nachdem die Karten zufällig angeordnet (gemischt) wurden müssen wir sie noch "umdrehen", das heißt unsichtbar machen. Dazu setzen wir die Hintergrundfarbe der Zelle auf die Schriftfarbe. Gleichzeitig setzen wir noch den Punktestand auf 0, geben diesen im Tabellenblatt aus und legen fest, dass die erste Karte noch nicht umgedreht wurde:

Da wir nun bereits globale Definitionen hinterlegt haben, sollten wir den gesamten Code der Funktion **SpielStarten** nochmals überarbeiten. Er könnte dann konkret so aussehen:

```
Function Starten()                                      'Funktion zum Starten:
Dim i                   As Integer                      'Allgemeine Zaehlvariable.
Dim iZufall             As Integer                      'Eine Zufallszahl.
Dim iZeile              As Integer                      'Aktuelle Zeilennummer.
Dim iSpalte             As Integer                      'Aktuelle Spaltennummer.
Dim stKarten            As String                       'Zeichen fuer Karteninhalte
Dim stKartenInhalt(MAX_PAARE * 2) As String             'Karteninhalte wie aufgelegt.
```

```
Randomize (Timer)                                   'Zufallsgenerator initialisieren.
Set oTabellenblatt = Sheets(TABELLENBLATT)          'Akt. Tabellenblatt festlegen.
bZug1 = False                                       'Erste Karte nicht aufgedeckt.
iPunkte = 0                                         'Punktestand ist Null.
Range("D2") = "Punkte: 000"                         'Punktestand ausgeben.
stKarten = "CD"                                     'Karteninhalte definieren.

stKarten = stKarten & stKarten                      'Karten verdoppeln.
For i = 1 To Len(stKarten)                          'Alle Karteninhalte zum
  stKartenInhalt(i) = Mid(stKarten, i, 1)           'Mischen vorbereiten bzw.
Next i                                              'ins Kartenfeld einlesen.

For i = 1 To MAX_PAARE * 2                          'Vorbereitetes Kartenfeld mischen
  iZufall = Int(Rnd() * MAX_PAARE * 2) + 1          'Jede Karte zumindest einmal
  stKartenInhalt(0) = stKartenInhalt(iZufall)       'vertauschen.
  stKartenInhalt(iZufall) = stKartenInhalt(i)
  stKartenInhalt(i) = stKartenInhalt(0)
Next i

i = 1                                               'Zaehlvariable ruecksetzen.
For iZeile = 1 To MAX_ZEILEN                        'Der Reihe nach die Zellen mit
  For iSpalte = 1 To MAX_SPALTEN                    'dem Inhalt der Karten befuellen:
    oTabellenblatt.Cells(iZeile, iSpalte).Value = stKartenInhalt(i): i = i + 1
    oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color = ROT 'Hintergrundfarbe ROT
    oTabellenblatt.Cells(iZeile, iSpalte).Font.Color = ROT     'und Schriftfarbe ROT
  Next iSpalte
Next iZeile
End Function                                        'Funktionsende.
```

Somit hätten wir das Memory-Spiel für den Spieleinsatz fertig aufgebaut. Jetzt müssen wir nur noch den Spielablauf programmieren.

Dazu legen wir uns die Prozedur **Worksheet_SelectionChange** im Excel Objekt **Tabelle1 (Memory)** an. Diese Prozedur wird jedes Mal dann aufgerufen, wenn eine neue Zelle ausgewählt bzw. selektiert wird. Wir wollen jene Karte aufdecken, welche wir anklicken. Dazu beschränken wir den Bereich auf das Spielfeld in welches wir einen Klick machen dürfen. Natürlich darf die Karte auch noch nicht aufgedeckt sein.

Zur Unterscheidung, ob die erste Karte bereits aufgedeckt ist, verwenden wir die globale Variable **bZug1** welche wir zuvor im Modul **Globales** definiert haben.

Ist die erste Karte noch nicht aufgedeckt, dann machen wir den Karteninhalt sichtbar indem wir die Hintergrundfarbe der angeklickten Zelle auf **WEISS** setzen. Zusätzlich merken wir uns die Position der Karte und deren Inhalt.

Ist die erste Karte bereits aufgeschlagen, dann lassen wir uns den Inhalt der 2. Karte anzeigen. Sind die beiden Karteninhalte unterschiedlich, dann lassen wir nach einer gewissen Zeit, in der sich der Spieler den Karteninhalt merken kann, die beiden Karteninhalte wieder unsichtbar werden, indem wir die Hintergrundfarbe der beiden Karten wieder auf die Schriftfarbe ändern.

Sind hingegen die Karteninhalte identisch, dann bleiben die Karten aufgedeckt. Wir erhöhen die Punkteanzahl und geben diese im Tabellenblatt aus. Dabei Kontrollieren wir noch, ob bereits alle Karten aufgedeckt sind. Wenn ja, dann ist den Spiel beendet.

Nachdem die zweite Karte umgedreht wurde merken wir uns für den nächsten Spielzug, dass die erste Karte noch nicht umgedreht wurde.

Die fertige Prozedur könnte dann so aussehen:

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)
Dim iZeile As Integer                                   'Angeklickte Zeilennummer.
Dim iSpalte As Integer                                  'Angeklickte Spaltennummer.

iZeile = Target.Row                                     'Aktuelle Zeile merken.
iSpalte = Target.Column                                 'aktuelle Spalte merken.
Set oTabellenblatt = Sheets(TABELLENBLATT)              'Akt. Tabellenblatt festlegen.

'Wenn Zeile und Spalte im Spielfeldbereich und die Karte nicht aufgedeckt ist:
If iZeile > 0 And iZeile <= MAX_ZEILEN And _
   iSpalte > 0 And iSpalte <= MAX_SPALTEN And _
   oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color <> WEISS Then

  If bZug1 = False Then                                 'Erste Karte nicht aufgedeckt:
    oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color = WEISS '1. Karte anzeigen.
    bZug1 = True                                        'Merken 1. Karte ist aufgedeckt.
    iZeileZug1 = iZeile                                 'Merken Zeile der Karte.
    iSpalteZug1 = iSpalte                               'Merken Spalte der Karte
    stInhaltZug1 = oTabellenblatt.Cells(iZeile, iSpalte).Value 'und Karteninhalt.
  Else                                                  '2. Karte wird umgedreht:
    oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color = WEISS '2. Karte anzeigen.
    'Wenn Inhalt der 1. Karte ungleich der 2. Karte, dann etwas warten und dann
    If stInhaltZug1 <> oTabellenblatt.Cells(iZeile, iSpalte).Value Then
      Sleep (750)                                       'beide Karten unsichtbar machen:
      oTabellenblatt.Cells(iZeileZug1, iSpalteZug1).Interior.Color = ROT
      oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color = ROT
    Else                                                '1. und 2. Karte sind gleich:
      iPunkte = iPunkte + 1                             'Punkte zaehlen und ausgeben.
      Range("D2") = "Punkte: " & Right("000" & Trim(Str(iPunkte)), 3)
      If iPunkte = MAX_PAARE Then                       'Wenn alle Karten aufgedeckt,
        MsgBox "Spielende"                              'dann Spielende.
      End If
    End If                                              'Nachdem 2. Karte aufgedeckt
    bZug1 = False                                       'wurde, merken dass 1. Karte
  End If                                                'nicht aufgedeckt wurde.
End If
End Sub                                                 'Prozedurende.
```

Diese Prozedur hat jedoch noch einen Schönheitsfehler: Sie wird auch dann aufgerufen, wenn man einen Doppelklick auf eine Karte macht bzw. wenn man über die Tastatur auf eine andere Zelle springt. Dieses Problem lösen wir folgendermaßen – zuerst das Doppelklicken deaktivieren – im Excel Objekt **Tabelle1 (Memory)**:

```
Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)
Cancel = True                                           'Doppelklick abbrechen.
End Sub
```

Die Steuerung über die Tastatur ändern wir über das Ereignis **OnKey**. Mit **OnKey** kann man das vordefinierte Verhalten von Tasten steuern. Man kann einer beliebigen Taste eine Funktion zuweisen, die Taste deaktivieren oder die ursprüngliche Hinterlegung (Bedeutung) wiederherstellen. Beispiele:

```
Application.OnKey "{ESC}", "Fluchttaste"                'Funktion zuweisen oder
Application.OnKey "{ESC}", ""                           'Taste deaktivieren, nix machen.
Application.OnKey "{ESC}"                               'Ursprung wieder herstellen.
```

Wir wollen nun alle Tasten, welche den Cursor bzw. den Fokus auf eine andere Zelle verschieben können, deaktivieren. Das soll gleich beim Öffnen der Excel-Datei geschehen.

Wir erinnern uns: Zuerst wird die Prozedur **Workook_Open** im Excel Objekt **DieseArbeitsmappe** ausgeführt. Dabei werden zunächst auch alle Funktionen die dort hinterlegt sind abgearbeitet. Erst danach wird noch die Prozedur **Workbook_Activate** ausgeführt.

Daher deaktivieren wir die Tasten in der Prozedur Workbook_Activate.

```
Private Sub Workbook_Activate()
Application.OnKey "{HOME}", ""          'POS1
Application.OnKey "{PGDN}", ""          'BILD-AB
Application.OnKey "{PGUP}", ""          'BILD-AUF
Application.OnKey "{END}", ""           'ENDE
Application.OnKey "{LEFT}", ""          'NACH-LINKS
Application.OnKey "{RIGHT}", ""         'NACH-RECHTS
Application.OnKey "{UP}", ""            'NACH-OBEN
Application.OnKey "{DOWN}", ""          'NACH-UNTEN
Application.OnKey "{TAB}", ""           'TAB
Application.OnKey "{DELETE}", ""        'ENTF
Application.OnKey "{CLEAR}", ""         'CLEAR
Application.OnKey "{BACKSPACE}", ""     'RÜCKTASTE
Application.OnKey "{ENTER}", ""         'EINGABETASTE
Application.OnKey "{RETURN}", ""        'RETURN-TASTE
Application.OnKey "~", ""               'EINGABETASTE (Ziffernblock)
End Sub                                 'Prozedurende.
```

Damit die Tasten in anderen geöffneten Excel-Dateien ihre normale Funktion beibehalten, müssen wir sie bei einem Dateiwechsel wieder aktivieren. Das erledigen wir in der Prozedur **Workbook_Deactivate**:

```
Private Sub Workbook_Deactivate()
Application.OnKey "{HOME}"              'POS1
Application.OnKey "{PGDN}"              'BILD-AB
Application.OnKey "{PGUP}"              'BILD-AUF
Application.OnKey "{END}"               'ENDE
Application.OnKey "{LEFT}"              'NACH-LINKS
Application.OnKey "{RIGHT}"             'NACH-RECHTS
Application.OnKey "{UP}"                'NACH-OBEN
Application.OnKey "{DOWN}"              'NACH-UNTEN
Application.OnKey "{TAB}"               'TAB
Application.OnKey "{DELETE}"            'ENTF
Application.OnKey "{CLEAR}"             'CLEAR
Application.OnKey "{BACKSPACE}"         'RÜCKTASTE
Application.OnKey "{ENTER}"             'EINGABETASTE
Application.OnKey "{RETURN}"            'RETURN-TASTE
Application.OnKey "~"                   'EINGABETASTE (Ziffernblock)
End Sub                                 'Prozedurende.
```

So das wär´s fürs Erste.

Speichern, schließen und Datei neu öffnen – der kleinstmögliche Spielspaß kann beginnen.

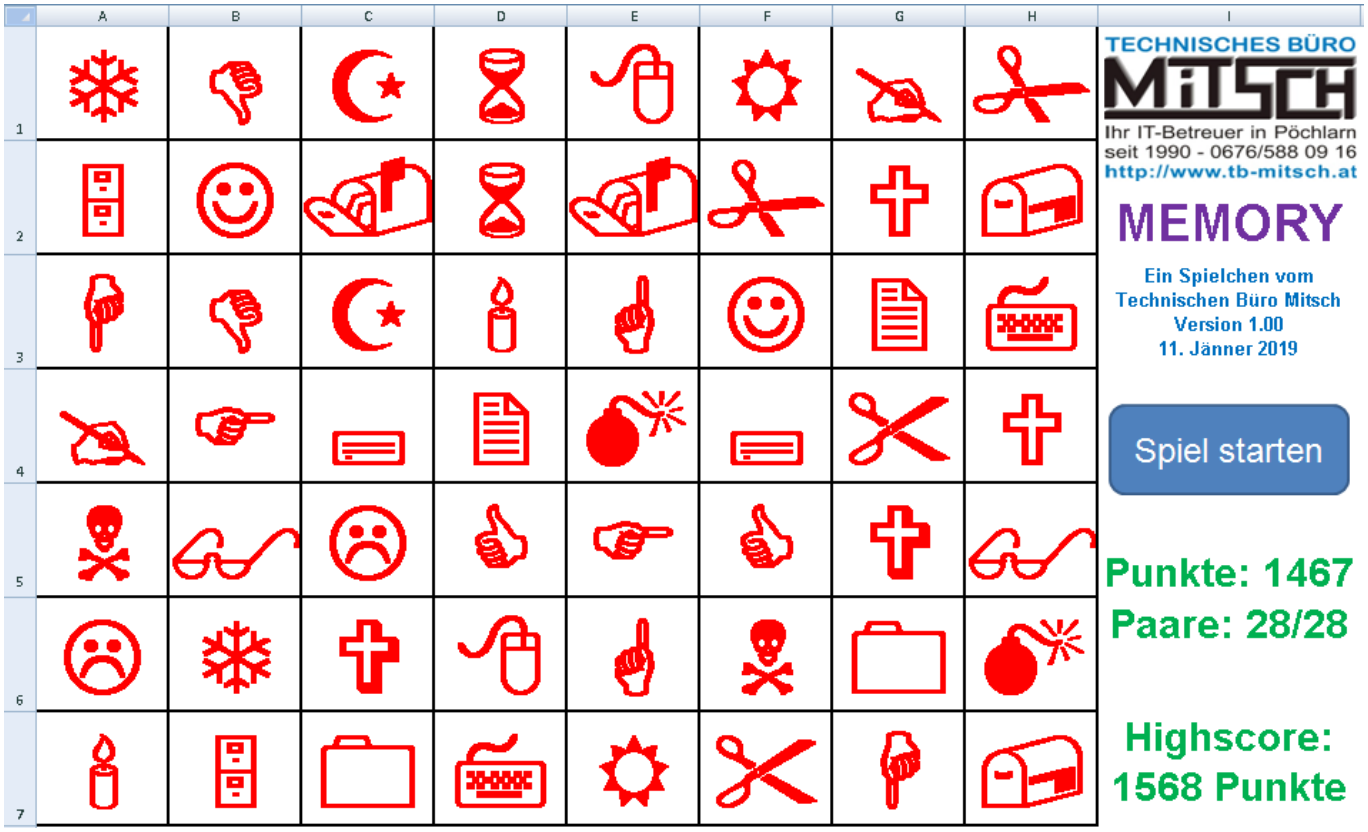
4.3.) Programmierung des vollständigen Spieles

Ab jetzt arbeiten wir in Mustervorlage der Excel-Datei welche wir zuvor auf **Memory.xlsm** umbenannt haben und erweitern die Vorlage zum richtigen Spiel.

Dabei gehen wir folgendermaßen vor. Wir öffnen gleichzeitig die beiden Excel-Dateien **Memory.xlsm** und **Memory_Entwurf.xlsm**. Dann kopieren wir über die Zwischenablage den Programmcode für die einzelnen Module aus dem Entwurf in unsere tatsächliche Spieldatei. Dabei vergessen wir auch nicht die Fehlerbehandlungsroutine zu erweitern und alles schön zu dokumentieren!

4.3.1.) Gestaltung des endgültigen Spielfeldes

Nun gestalten wir ein neues Spielfeld für 8 x 7 Karten (28 Paare). Als Schriftart verwenden wir den Font Wingdings (ist standardmäßig in Windows vorhanden). Als sinnvolle Zeichen in diesem Font kann man den ASCII-Code von 33 bis 93 verwenden (61 Zeichen). Mein Vorschlag könnte eventuell so aussehen:



Wer will kann auch die Funktion **SpielfeldZeichnen** (im Modul **Funktionen** hinterlegt) verwenden um das aktuell ausgewählte Tabellenblatt automatisch mit meinen Einstellungen zu gestalten:

```
Public Function SpielfeldZeichnen()
'*************************************************************************/
'*      INTERNES UNTERPROGRAMM - INTERNE FUNKTION                        */
'*************************************************************************/
'*----------------------------------------------------------------------*/
'*                                                                       */
'*      FUNKTION:       SpielfeldZeichnen                                */
'*                                                                       */
'*      BESCHREIBUNG:   Nur zum formatieren des Tabellenblattes: Memory  */
```

```
'*                                                                            */
'*      RETURN          nichts                                                */
'*                                                                            */
'*      PARAMETER:      keine                                                 */
'*                                                                            */
'*----------------------------------------------------------------------------*/
Dim stBereich           As String                   'Bereich des Spielfeldes.

stBereich = "A1:H7"                                 'Fuer 8 x 6 Karten.
Range(stBereich).Select                             'Bereich im aktuell ausgewaehlten
                                                    'Tabellenblatt selektieren.
Selection.ColumnWidth = 17                          'Spaltenbreite.
Selection.RowHeight = 80                            'Zeilenhoehe.
Selection.HorizontalAlignment = xlCenter            'Horizontale Ausrichtung.
Selection.VerticalAlignment = xlCenter              'Vertikale Ausrichtung.
Selection.Interior.Color = 255                      'Hintergrundfarbe = ROT.
Selection.Font.Color = 255                          'Schriftfarbe = ROT.
Selection.Font.Name = "Wingdings"                   'Schriftsatz, Font.
Selection.Font.Size = 70                            'Schriftgroesse.
Selection.Font.Bold = True                          'Fettschrift = JA.
Selection.Borders(xlDiagonalDown).LineStyle = xlNone'Eine dicke, schwarze
Selection.Borders(xlDiagonalUp).LineStyle = xlNone  'Rahmenlinie ueber alle
With Selection.Borders(xlEdgeLeft)                  'selektierte Zellen
    .LineStyle = xlContinuous                       'zeichnen lassen.
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .Weight = xlMedium
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .Weight = xlMedium
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .Weight = xlMedium
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .Weight = xlMedium
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .Weight = xlMedium
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .Weight = xlMedium
End With
Range("A1").Select                                  'Zelle A1 selektieren.
End Function                                        'Funktionsende.
```

4.3.2.) Programmcode aus DieseArbeitsmappe

Hier gleich der vollständige Programmcode. Gelb hinterlegt sind jene Zeilen, welche zusätzlich zum Code aus der Mustervorlage erweitert wurden.

```
'*********************************************************************************/
'*                                                                               */
'*                                  \\\\|////                                    */
'*                                  \\ _ _ //                                    */
'*                                   ( o o )                                     */
'*================================oOOo-(_)-oOOo==================================*/
'* WICHTIG: Dieser Code benoetigt zusaetzliche Verweise auf die ActiveX-dll      */
'* Klick: Extras > Verweise > Visual Basic for Applications                      */
'* Klick: Extras > Verweise > Microsoft Excel 12.0 Objects Library               */
'* Klick: Extras > Verweise > Microsoft Office 12.0 Objects Library              */
'* Klick: Extras > Verweise > OLE Automation                                     */
'*=======================================Oooo====================================*/
'*                                  oooO  (  )                                   */
'*                                  (  )  ) /                                    */
'*                                   \ ( (_/                                     */
'*                                    \_)                                        */
'*                                                                               */
'*********************************************************************************/
'*********************************************************************************/
'*                                                                               */
'*      PROGRAMM:       Memory                                                   */
'*      PROJEKT:        Spiele programmieren in Excel                            */
'*                                                                               */
'*      MODULNAME:      Diese Arbeitsmappe                                       */
'*      ARCHIVIERT:     -                                                        */
'*                                                                               */
'*      VERSION:        V  01.00                                                 */
'*      VERSIONSDATUM:  11 Jan 2019 12:00:00                                     */
'*                                                                               */
'*      BEARBEITER:     Ing. Harald Mitsch, TBM                                  */
'*                                                                               */
'*===============================================================================*/
'*                                                                               */
'*      (C) 2019 by TBM-Technisches Buero Mitsch   Elektrotechn. Planungen       */
'*      Lizenz: FREEWARE                           und ET - Installationen       */
'*      Josefgasse 6, 3380 Poechlarn               EDV - Selfmade Software       */
'*      Tel.:02757 / 46 56 bzw. 0676 /588 09 16    allgemeines Zeichenbüro       */
'*                                                                               */
'*===============================================================================*/
'*                                                                               */
'*      BESCHREIBUNG:   Behandelt Funktionen die beim Oeffnen oder beim          */
'*                      Schliessen der Excel-Datei automatisch ausgefuehrt       */
'*                      werden.                                                  */
'*                                                                               */
'*      NEBENEFFEKTE:   -                                                        */
'*                                                                               */
'*      GRENZEN:        -                                                        */
'*                                                                               */
'*===============================================================================*/
'*                                                                               */
'*      DEKLARIERTE FUNKTIONEN:   keine                                          */
'*                                                                               */
'*      EXPORTIERTE FUNKTIONEN:   keine                                          */
'*                                                                               */
'*      IMPORTIERTE FUNKTIONEN:   SpielAnsichtAktualisieren                      */
'*                                SpielStarten                                   */
'*                                                                               */
'*===============================================================================*/
'*                                                                               */
'*      EXPORTIERTE VARIABLEN:    keine                                          */
'*                                                                               */
```

```
'*      IMPORTIERTE VARIABLEN:    alles aus Modul: Globales                 */
'*                                                                          */
'*==========================================================================*/
'*                                                                          */
'*      INTERNE FUNKTIONEN:       Workbook_Open                             */
'*                                Workbook_BeforeClose                      */
'*                                Workbook_Activate                         */
'*                                Workbook_Deactivate                       */
'*                                                                          */
'*==========================================================================*/
'*                                                                          */
'*      AENDERUNGEN: - Modification History:                                */
'*                                                                          */
'*      V 01.00     11.01.2019    Mitsch Harald, TBM, Initial Version       */
'*                                                                          */
'****************************************************************************/
Option Explicit                                 'Alle Variablen muessen
                                                'explizit angegeben werden.

Private Sub Workbook_Open()
'****************************************************************************/
'*      INTERNES UNTERPROGRAMM - INTERNE FUNKTION - PROZEDUR                */
'****************************************************************************/
'*--------------------------------------------------------------------------*/
'*                                                                          */
'*      FUNKTION:       Workbook_Open                                       */
'*                                                                          */
'*      BESCHREIBUNG:   Diese Funktion wird sofort aufgerufen nachdem die   */
'*                      Excel-Datei geoeffnet wurde. Hier werden all jene   */
'*                      Einstellungen festgelegt, welche im gesamten Excel  */
'*                      Gueltigkeit haben.                                  */
'*                      Zum Schluss erfolgt der eigentliche Programmstart:  */
'*                                                                          */
'*                      Soll diese Prozedur nicht automatisch gestartet     */
'*                      werden, so ist beim Oeffnen der Excel-Datei die     */
'*                      SHIFT-Taste (Grossschreibtaste) gedrueckt zu        */
'*                      halten.                                             */
'*                                                                          */
'*      RETURN          nichts                                              */
'*                                                                          */
'*      PARAMETER:      keine                                               */
'*                                                                          */
'*--------------------------------------------------------------------------*/
On Error GoTo Fehler                            'Bei Fehler zur Fehlerbehandlung.
                                                'Wenn gewuenscht, dann
Application.WindowState = xlMaximized           'Excel Fenster immer maximieren.

lExcelFensterGroesse = Application.WindowState      'Excel Fenstergroesse auslesen.
bAnzeigenGanzerBildschirm = Application.DisplayFullScreen    'Merken der aktuellen
bAnzeigenTabellenblaetter = ActiveWindow.DisplayWorkbookTabs 'Anzeigeeinstellungen
bAnzeigenVertikaleSchiebeleiste = ActiveWindow.DisplayVerticalScrollBar  'von Excel
bAnzeigenHorizontaleSchiebeleiste = ActiveWindow.DisplayHorizontalScrollBar 'um sie
bAnzeigenZeilenSpaltenBezeichnung = ActiveWindow.DisplayHeadings 'spaeter wieder-
bAnzeigenGitternetzlinien = ActiveWindow.DisplayGridlines    'herstellen zu koennen.
lTabellenblattFensterGroesse = ActiveWindow.WindowState
iTabellenblattZoom = ActiveWindow.Zoom

ActiveWorkbook.EnableAutoRecover = False        'Auto-Wiederherstellung fuer
                                                'diese Excel-Datei ausschalten.

With Application                                'Ab Excel 2007 sollte zusaetzlich
  .DecimalSeparator = ","                       'dieser Codeteil aktiviert werden,
  .ThousandsSeparator = "."                     'damit die Dezimal- und Tausender-
  .UseSystemSeparators = False                  'Trennzeichen nicht aus dem
End With                                        'Betriebssystem uebernommen werden.

                                                'Idealer Zeitpunkt um auch den
```

```
Randomize (Date + Time)                                 'Zufallsgenerator zu initialisieren.
'iZufall = Int((6 * Rnd) + 1)                           'Beispiel fuer einen Wuerfel.

Call SpielAnsichtAktualisieren(SPIELANSICHT)            'Umschalten auf Entwickleransicht
                                                        'oder auf Spielansicht, je nach
                                                        'Definition im Modul Globales.

'*-------------------------------------------------------------------------*/
Call SpielStarten                                       'Startprogramm aufrufen.
'*-------------------------------------------------------------------------*/

                                                        'Fehlerbehandlungsroutine:
Beenden:                                                '=========================
  Application.Cursor = xlDefault                        'Sanduhr ausblenden.
  Exit Sub                                              'Prozedur abbrechen.

Fehler:                                                 'Fehlermarke - Fehlerbehandlung:
  MsgBox Err.Description                                'Fehlermeldung ausgeben.
  Resume Beenden                                        'Bei Funktionsende weitermachen.
  Resume                                                'Nur fuer Testzwecke hinterlegt.
End Sub                                                 'Prozedurende.


Private Sub Workbook_BeforeClose(Cancel As Boolean)
'*************************************************************************/
'*      INTERNES UNTERPROGRAMM - INTERNE FUNKTION - PROZEDUR              */
'*************************************************************************/
'*-------------------------------------------------------------------------*/
'*                                                                         */
'*      FUNKTION:       Workbook_BeforeClose                               */
'*                                                                         */
'*      BESCHREIBUNG:   Diese Prozedur wird vor den Schliessen der Excel-  */
'*                      Datei bzw. vor dem Beenden von Excel aufgerufen.   */
'*                      Wird in dieser Prozedur der Parameter: Cancel      */
'*                      auf True gesetzt, so wird das Schliessen bzw. das  */
'*                      Beenden abgebrochen - Excel-Datei bleibt geoeffnet.*/
'*                                                                         */
'*      RETURN          nichts                                             */
'*                                                                         */
'*      PARAMETER:      Boolean .......... True = Funktion abbrechen       */
'*                                                                         */
'*-------------------------------------------------------------------------*/
On Error GoTo Fehler                                    'Bei Fehler zur Fehlerbehandlung.
If bFluchttaste = True Then                             'Wurde die Escape-Taste gedrueckt,
  Application.Quit                                      'dann Excel beenden.
Else
  If AUTOSPEICHERN = JA Then                            'Je nach Definition im Modul Globales
    Application.DisplayAlerts = False                   'die aktuelle Arbeitsmappe automatisch
    ActiveWorkbook.Save                                 'speichern oder alle Aenderungen
    Application.DisplayAlerts = True                    'verwerfen. Dabei die Ausgabe von
                                                        'Fehler- und Hinweismeldungen
deaktivieren.
  ElseIf FRAGESPEICHERN = NEIN Then                     'Alle Aenderungen verwerfen:
    ThisWorkbook.Saved = True                           '(Excel mitteilen, dass Datei gespeichert
  Else                                                  'ist - auch wenn es nicht stimmt - daher
    'In diesem Else-Zweig stellt Excel die Frage       'versucht Excel gar nicht zu speichern.)
    'ob die Aenderungen gespeichert werden sollen.
  End If
                                                        'Nach Programmabsturz kann dieser
  Application.EnableEvents = True                       'Befehl sehr hilfreich sein!
End If


                                                        'Fehlerbehandlungsroutine:
Beenden:                                                '=========================
  Application.Cursor = xlDefault                        'Sanduhr ausblenden.
  Exit Sub                                              'Prozedur abbrechen.
```

```
Fehler:                                                 'Fehlermarke - Fehlerbehandlung:
  MsgBox Err.Description                                'Fehlermeldung ausgeben.
  Resume Beenden                                        'Bei Funktionsende weitermachen.
  Resume                                                'Nur fuer Testzwecke hinterlegt.
End Sub                                                 'Prozedurende.


Private Sub Workbook_Activate()
'*************************************************************************/
'*      INTERNES UNTERPROGRAMM - INTERNE FUNKTION - PROZEDUR             */
'*************************************************************************/
'*-----------------------------------------------------------------------*/
'*                                                                       */
'*      FUNKTION:       Workbook_Activate                                */
'*                                                                       */
'*      BESCHREIBUNG:   Diese Prozedur wird aufgerufen, wenn man aus einer  */
'*                      anderen geoeffneten Excel-Datei zu dieser zurueck-  */
'*                      wechselt. Egal ob ein Excel gestartet wurde und     */
'*                      damit mehrere Dateien geoeffnet sind oder ob Excel  */
'*                      mehrmals gestartet wurde und mit jedem Excel nur    */
'*                      eine Datei geoeffnet wurde.                      */
'*                                                                       */
'*                      Dieses Prozedur soll dafuer sorgen, dass die        */
'*                      Anzeigeeinstellungen fuer diese Datei automatisch   */
'*                      wiederhergestellt werden.                        */
'*                                                                       */
'*      RETURN          nichts                                           */
'*                                                                       */
'*      PARAMETER:      keine                                            */
'*                                                                       */
'*-----------------------------------------------------------------------*/
On Error GoTo Fehler                                    'Bei Fehler zur Fehlerbehandlung.
Application.OnKey "{ESC}", "Fluchttaste"                'Funktionszuweisung an ESC-Taste
                                                        'wieder aktivieren.
Application.OnKey "{HOME}", ""                          'POS1
Application.OnKey "{PGDN}", ""                          'BILD-AB
Application.OnKey "{PGUP}", ""                          'BILD-AUF
Application.OnKey "{END}", ""                           'ENDE
Application.OnKey "{LEFT}", ""                          'NACH-LINKS
Application.OnKey "{RIGHT}", ""                         'NACH-RECHTS
Application.OnKey "{UP}", ""                            'NACH-OBEN
Application.OnKey "{DOWN}", ""                          'NACH-UNTEN
Application.OnKey "{TAB}", ""                           'TAB
Application.OnKey "{DELETE}", ""                        'ENTF
Application.OnKey "{CLEAR}", ""                         'CLEAR
Application.OnKey "{BACKSPACE}", ""                     'RÜCKTASTE
Application.OnKey "{ENTER}", ""                         'EINGABETASTE
Application.OnKey "{RETURN}", ""                        'RETURN-TASTE
Application.OnKey "~", ""                               'EINGABETASTE (Ziffernblock)

Call SpielAnsichtAktualisieren(SPIELANSICHT)            'Umschalten auf Entwickleransicht
                                                        'oder auf Spielansicht, je nach
                                                        'Definition im Modul Globales.
                                                        'Wenn gewuenscht, dann
Application.WindowState = xlMaximized                   'Excel Fenster immer maximieren.

                                                        'Fehlerbehandlungsroutine:
Beenden:                                                '=========================
  Application.Cursor = xlDefault                        'Sanduhr ausblenden.
  Exit Sub                                              'Prozedur abbrechen.

Fehler:                                                 'Fehlermarke - Fehlerbehandlung:
  MsgBox Err.Description                                'Fehlermeldung ausgeben.
  Resume Beenden                                        'Bei Funktionsende weitermachen.
  Resume                                                'Nur fuer Testzwecke hinterlegt.
End Sub                                                 'Prozedurende.
```

```
Private Sub Workbook_Deactivate()
'*************************************************************************/
'*      INTERNES UNTERPROGRAMM - INTERNE FUNKTION - PROZEDUR              */
'*************************************************************************/
'*-----------------------------------------------------------------------*/
'*                                                                       */
'*      FUNKTION:       Workbook_Deactivate                              */
'*                                                                       */
'*      BESCHREIBUNG:   Diese Prozedur wird aufgerufen, wenn man aus dieser */
'*                      Excel-Datei zu einer anderen geoeffneten Excel-Datei*/
'*                      wechselt. Egal ob ein Excel gestartet wurde und  */
'*                      damit mehrere Dateien geoeffnet sind oder ob Excel */
'*                      mehrmals gestartet wurde und mit jedem Excel nur */
'*                      eine Datei geoeffnet wurde.                      */
'*                                                                       */
'*                      Dieses Prozedur soll dafuer sorgen, dass die     */
'*                      Anzeigeeinstellungen fuer diese Datei NICHT in die */
'*                      anderen Excel-Datei uebernommen wird.            */
'*                                                                       */
'*      RETURN          nichts                                           */
'*                                                                       */
'*      PARAMETER:      keine                                            */
'*                                                                       */
'*-----------------------------------------------------------------------*/
On Error GoTo Fehler                                'Bei Fehler zur Fehlerbehandlung.
Application.OnKey "{ESC}"                           'Funktionszuweisung an ESC-Taste
                                                    'wieder entfernen.
Application.OnKey "{HOME}"                          'POS1
Application.OnKey "{PGDN}"                          'BILD-AB
Application.OnKey "{PGUP}"                          'BILD-AUF
Application.OnKey "{END}"                           'ENDE
Application.OnKey "{LEFT}"                          'NACH-LINKS
Application.OnKey "{RIGHT}"                         'NACH-RECHTS
Application.OnKey "{UP}"                            'NACH-OBEN
Application.OnKey "{DOWN}"                          'NACH-UNTEN
Application.OnKey "{TAB}"                           'TAB
Application.OnKey "{DELETE}"                        'ENTF
Application.OnKey "{CLEAR}"                         'CLEAR
Application.OnKey "{BACKSPACE}"                     'RÜCKTASTE
Application.OnKey "{ENTER}"                         'EINGABETASTE
Application.OnKey "{RETURN}"                        'RETURN-TASTE
Application.OnKey "~"                               'EINGABETASTE (Ziffernblock)

If bFluchttaste = False Then                        'Wenn Fluchttaste nicht gedrückt:
  Application.DisplayFullScreen = False             'Vollbildmodus ausschalten.
  Application.WindowState = lExcelFensterGroesse    'Excel Fenstergroesse wiederherstellen.
End If
                                                    'Fehlerbehandlungsroutine:
Beenden:                                            '========================
  Application.Cursor = xlDefault                    'Sanduhr ausblenden.
  Exit Sub                                          'Prozedur abbrechen.

Fehler:                                             'Fehlermarke - Fehlerbehandlung:
  MsgBox Err.Description                            'Fehlermeldung ausgeben.
  Resume Beenden                                    'Bei Funktionsende weitermachen.
  Resume                                            'Nur fuer Testzwecke hinterlegt.
End Sub                                             'Prozedurende.
```

4.3.3.) Programmcode aus Tabelle1 (Memory)

Vorab möchte ich noch darauf hinweisen, dass ich die Zellen im Tabellenblatt Memory, wo der Highscore gespeichert wird und die Punkte, die Paare und der Highscore ausgegeben wird mit Namen versehen habe: HIGHSCORE, PUNKTE, PAARE und HIGHSCORE_TEXT. Wobei zusätzlich die Schriftfarbe in der Zelle, wo der Highscore abgespeichert wird, auf die Hintergrundfarbe geändert wurde. Der Wert des Highscores ist somit unsichtbar.

Hier gleich der vollständige Programmcode. Er wurde vollständig neu erstellt bzw. aus dem Entwurf übernommen.

```
'*************************************************************************/
'*                                                                       */
'*      PROGRAMM:       Memory                                           */
'*      PROJEKT:        Spiele programmieren in Excel                    */
'*                                                                       */
'*      MODULNAME:      Tabelle1 (Memory)                                */
'*      ARCHIVIERT:     -                                                */
'*                                                                       */
'*      VERSION:        V  01.00                                         */
'*      VERSIONSDATUM:  11 Jan 2019 12:00:00                             */
'*                                                                       */
'*      BEARBEITER:     Ing. Harald Mitsch, TBM                          */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      (C) 2019 by TBM-Technisches Buero Mitsch   Elektrotechn. Planungen  */
'*      Lizenz: FREEWARE                           und ET - Installationen  */
'*      Josefgasse 6, 3380 Poechlarn               EDV - Selfmade Software  */
'*      Tel.:02757 / 46 56 bzw. 0676 /588 09 16    allgemeines Zeichenbüro  */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      BESCHREIBUNG:   Behandelt Funktionen die direkt aus dem Tabellenlatt*/
'*                      aufgerufen werden.                               */
'*                                                                       */
'*      NEBENEFFEKTE:   -                                                */
'*                                                                       */
'*      GRENZEN:        -                                                */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      DEKLARIERTE FUNKTIONEN:   keine                                  */
'*                                                                       */
'*      EXPORTIERTE FUNKTIONEN:   keine                                  */
'*                                                                       */
'*      IMPORTIERTE FUNKTIONEN:   keine                                  */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      EXPORTIERTE VARIABLEN:    keine                                  */
'*                                                                       */
'*      IMPORTIERTE VARIABLEN:    alles aus Modul: Globales              */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      INTERNE FUNKTIONEN:       Worksheet_SelectionChange              */
'*                                Worksheet_BeforeDoubleClick            */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      AENDERUNGEN: - Modification History:                             */
'*                                                                       */
'*      V 01.00     11.01.2019    Mitsch Harald, TBM, Initial Version    */
```

```
'*                                                                            */
'******************************************************************************/
Option Explicit                                 'Alle Variablen muessen
                                                'explizit angegeben werden.

Private Sub Worksheet_SelectionChange(ByVal Target As Range)
'******************************************************************************/
'*      INTERNES UNTERPROGRAMM - INTERNE FUNKTION - PROZEDUR                  */
'******************************************************************************/
'*----------------------------------------------------------------------------*/
'*                                                                            */
'*      FUNKTION:       Worksheet_SelectionChange                             */
'*                                                                            */
'*      BESCHREIBUNG:   Diese Prozedur wird aufgerufen, wenn in diesem        */
'*                      Tabellenblatt der Cursor bzw. Fokus von einer Zelle   */
'*                      auf eine andere Zelle springt bzw. wenn eine neue     */
'*                      Zelle selektiert oder ausgewaehlt wird.               */
'*                                                                            */
'*      RETURN          nichts                                                */
'*                                                                            */
'*      PARAMETER:      keine                                                 */
'*                                                                            */
'*----------------------------------------------------------------------------*/
Dim iZeile              As Integer              'Angeklickte Zeilennummer.
Dim iSpalte             As Integer              'Angeklickte Spaltennummer.
Dim iPunktebewertung    As Integer              'Ergebnis Punkteberechnung.

On Error GoTo Fehler                            'Bei Fehler zur Fehlermarke.

iZeile = Target.Row                             'Aktuelle Zeile merken.
iSpalte = Target.Column                         'aktuelle Spalte merken.

Set oTabellenblatt = Sheets(TABELLENBLATT)      'Akt. Tabellenblatt festlegen.
oTabellenblatt.Cells(iZeile, iSpalte).Select

'Wenn Zeile und Spalte im Spielfeldbereich und die Karte nicht aufgedeckt ist:
If iZeile > 0 And iZeile <= MAX_ZEILEN And _
   iSpalte > 0 And iSpalte <= MAX_SPALTEN And _
   oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color <> WEISS Then

  If bZug1 = False Then                         'Erste Karte nicht aufgedeckt:
    oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color = WEISS '1. Karte anzeigen.
    bZug1 = True                                'Merken 1. Karte ist aufgedeckt.
    iZeileZug1 = iZeile                         'Merken Zeile der Karte.
    iSpalteZug1 = iSpalte                       'Merken Spalte der Karte
    stInhaltZug1 = oTabellenblatt.Cells(iZeile, iSpalte).Value 'und Karteninhalt.
  Else                                          '2. Karte wird umgedreht:
    oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color = WEISS '2. Karte anzeigen.
    'Wenn Inhalt der 1. Karte ungleich der 2. Karte, dann etwas warten und dann
    If stInhaltZug1 <> oTabellenblatt.Cells(iZeile, iSpalte).Value Then
      Sleep (750)                               'beide Karten unsichtbar machen:
      oTabellenblatt.Cells(iZeileZug1, iSpalteZug1).Interior.Color = ROT
      oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color = ROT
      iFehlversuche = iFehlversuche + 1         'Fehlversuche erhoehen.
    Else                                        '1. und 2. Karte sind gleich:
      iPunktebewertung = MAX_PAARE * 2 - iFehlversuche
      If iPunktebewertung < 0 Then iPunktebewertung = 0
      iPunkte = iPunkte + iPunktebewertung      'Punkte zaehlen und ausgeben.
      iPaare = iPaare + 1                       'Anzahl der Paare erhoehen.
      iFehlversuche = 0                         'Anzahl Fehlversuche ruecksetzen.
      Range("PUNKTE") = "Punkte: " & Right("0000" & Trim(Str(iPunkte)), 4)
      Range("PAARE") = "Paare: " & Right("00" & Trim(Str(iPaare)), 2) & _
                       "/" & Right("00" & Trim(Str(MAX_PAARE)), 2)
      If iPunkte > iHighscore Then              'Wenn Highscore ueberschritten,
        iHighscore = iPunkte                    'dann neuen Highscore speichern.
        iNeuerHighscore = True                  'Merken neuer Highscore erreicht.
        Range("HIGHSCORE") = iHighscore         'Neuen Highscore ausgeben.
```

```
        Range("HIGHSCORE_TEXT") = "Highscore:" & vbCrLf & Right("0000" & _
                                  Trim(Str(iHighscore)), 4) & " Punkte"
      End If


      If iPaare = MAX_PAARE Then                    'Wenn alle Karten aufgedeckt
        If iNeuerHighscore = True Then              'und neuer Highscore:
          MsgBox "Spielende" & vbCrLf & vbCrLf & "!!! NEUER HIGHSCORE !!!", _
                 vbOKOnly + vbInformation + vbDefaultButton1, PROGRAMMNAME
        Else                                        'Normales Spielende:
          MsgBox "Spielende", vbOKOnly + vbInformation + vbDefaultButton1, PROGRAMMNAME
        End If
      End If                                        'dann Meldungsausgabe: Spielende.

    End If                                          'Nachdem 2. Karte aufgedeckt
    bZug1 = False                                   'wurde, merken dass 1. Karte

  End If                                            'nicht aufgedeckt wurde.
End If
                                                    'Fehlerbehandlungsroutine:
Beenden:                                            '=========================
  Application.Cursor = xlDefault                    'Sanduhr ausblenden.
  Exit Sub                                          'Prozedur abbrechen.

Fehler:                                             'Fehlermarke - Fehlerbehandlung:
  MsgBox Err.Description                            'Fehlermeldung ausgeben.
  Resume Beenden                                    'Bei Funktionsende weitermachen.
  Resume                                            'Nur fuer Testzwecke hinterlegt.
End Sub                                             'Prozedurende.


Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)
'*************************************************************************/
'*      INTERNES UNTERPROGRAMM - INTERNE FUNKTION - PROZEDUR             */
'*************************************************************************/
'*-----------------------------------------------------------------------*/
'*                                                                       */
'*      FUNKTION:       Worksheet_BeforeDoubleClick                      */
'*                                                                       */
'*      BESCHREIBUNG:   Diese Prozedur wird aufgerufen, wenn man in diesem  */
'*                      Tabellenblatt einen Doppelklick macht.           */
'*                      Wird in dieser Prozedur der Parameter: Cancel    */
'*                      auf True gesetzt, so wird das Doppelklicken      */
'*                      abgebrochen.                                     */
'*                                                                       */
'*      RETURN          nichts                                           */
'*                                                                       */
'*      PARAMETER:      keine                                            */
'*                                                                       */
'*-----------------------------------------------------------------------*/
On Error GoTo Fehler                                'Bei Fehler zur Fehlermarke.

Cancel = True                                       'Doppelklick abbrechen.


                                                    'Fehlerbehandlungsroutine:
Beenden:                                            '=========================
  Application.Cursor = xlDefault                    'Sanduhr ausblenden.
  Exit Sub                                          'Prozedur abbrechen.

Fehler:                                             'Fehlermarke - Fehlerbehandlung:
  MsgBox Err.Description                            'Fehlermeldung ausgeben.
  Resume Beenden                                    'Bei Funktionsende weitermachen.
  Resume                                            'Nur fuer Testzwecke hinterlegt.
End Sub                                             'Prozedurende.
```

4.3.4.) Programmcode aus Modul Globales

Hier gleich der vollständige Programmcode. Geändertes und neues ist gelb hinterlegt.

```
'*************************************************************************/
'*                                                                       */
'*      PROGRAMM:       Memory                                           */
'*      PROJEKT:        Spiele programmieren in Excel                    */
'*                                                                       */
'*      MODULNAME:      Globales                                         */
'*      ARCHIVIERT:     -                                                */
'*                                                                       */
'*      VERSION:        V  01.00                                         */
'*      VERSIONSDATUM:  11 Jan 2019 12:00:00                             */
'*                                                                       */
'*      BEARBEITER:     Ing. Harald Mitsch, TBM                          */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      (C) 2019 by TBM-Technisches Buero Mitsch   Elektrotechn. Planungen  */
'*      Lizenz: FREEWARE                           und ET - Installationen  */
'*      Josefgasse 6, 3380 Poechlarn               EDV - Selfmade Software  */
'*      Tel.:02757 / 46 56 bzw. 0676 /588 09 16    allgemeines Zeichenbüro  */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      BESCHREIBUNG:   Definiert alle globalen Konstanten und Variablen.   */
'*                                                                       */
'*      NEBENEFFEKTE:   -                                                */
'*                                                                       */
'*      GRENZEN:        -                                                */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      DEKLARIERTE FUNKTIONEN:   Sleep                                  */
'*                                GetAsyncKeyState                       */
'*                                                                       */
'*      EXPORTIERTE FUNKTIONEN:   keine                                  */
'*                                                                       */
'*      IMPORTIERTE FUNKTIONEN:   keine                                  */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      EXPORTIERTE VARIABLEN:    alles as Deklarationsbereich           */
'*                                                                       */
'*      IMPORTIERTE VARIABLEN:    keine                                  */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      INTERNE FUNKTIONEN:       keine                                  */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      AENDERUNGEN: - Modification History:                             */
'*                                                                       */
'*      V 01.00     11.01.2019    Mitsch Harald, TBM, Initial Version    */
'*                                                                       */
'*************************************************************************/
Option Explicit                                     'Alle Variablen muessen
                                                    'explizit angegeben werden.
'-----------------------------------------------------------------------/
'-  Deklarierte API-Funktionen fuer 64 Bit Office Versionen             -/
'-----------------------------------------------------------------------/
#If VBA7 Then                                       'Fuer 64 Bit Excel
  Public Declare PtrSafe Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As LongPtr)
```

```
  Public Declare PtrSafe Function GetAsyncKeyState Lib "user32" (ByVal vKey As Long) As
Integer
'-----------------------------------------------------------------------------/
'-  Deklarierte API-Funktionen fuer 32 Bit Office Versionen                   -/
'-----------------------------------------------------------------------------/
#Else                                                   'Fuer 32 Bit Excel
  Public Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
                                                        'Fuer Sleep-Funktion.
  Public Declare Function GetAsyncKeyState Lib "user32" (ByVal vKey As Long) As Integer
                                                        'Fuer Tastendruckabfrage.
#End If

'-----------------------------------------------------------------------------/
'-  Globale Konstanten und Variablen - fuer alle Spiele benoetigt             -/
'-----------------------------------------------------------------------------/
Global Const PROGRAMMNAME = "Memory"                    'Name des Programmes.
                                                        'Name des Programmierers:
Global Const PROGRAMMAUTOR = "© 2019 by TBM - Technisches Büro Mitsch" '(c) = Asc(184)
Global Const TABELLENBLATT = "Memory"                   'Tabellenblatt fuer Spielfeld.
Global oTabellenblatt                       As Object   'Variable fuer Tabellenblatt.

Global Const JA = True                                  'Konstanten zur besseren Lesbarkeit
Global Const NEIN = False                               'des Programmcodes.

                                                        'Verhalten beim STARTEN:
'Global Const SPIELANSICHT = NEIN                       'Waehrend der Spielentwicklung.
Global Const SPIELANSICHT = JA                          'Nach Fertigstellung des Spiels.
Global Const ZOOMFAKTOR = 100                           'Gewuenschter Zoom-Faktor (100 %)

                                                        'Verhalten beim BENDEN:
'Global Const AUTOSPEICHERN = JA                       'Beim Beenden autom. alles speichern.
Global Const AUTOSPEICHERN = NEIN                       'Aenderungen automatisch Verwerfen
Global Const FRAGESPEICHERN = NEIN                      'wenn auch FRAGESPEICHERN = NEIN oder
'Global Const FRAGESPEICHERN = JA                      'zusaetzliche Nachfrage ob Speichern.

                                                        'Zum Merken der ANZEIGE-Einstellungen:
Global lTabellenblattFensterGroesse      As Long        'Application.WindowState
                                                        'Tabellenblatt Fenstergroesse.
Global bAnzeigenGanzerBildschirm         As Boolean     'Application.DisplayFullScreen
                                                        'Ganzer Bildschirm anzeigen.
Global bAnzeigenTabellenblaetter         As Boolean     'ActiveWindow.DisplayWorkbookTabs
                                                        'Tabellenblatt-Register anzeigen.
Global bAnzeigenVertikaleSchiebeleiste   As Boolean     'ActiveWindow.DisplayVerticalScrollBar
                                                        'Schiebeleiste rechts anzeigen.
Global bAnzeigenHorizontaleSchiebeleiste As Boolean     'ActiveWindow.DisplayHorizontalScrollBar
                                                        'Schiebeleiste unten anzeigen.
Global bAnzeigenZeilenSpaltenBezeichnung As Boolean     'ActiveWindow.DisplayHeadings
                                                        'Zeilen und Spaltenbeschriftung.
Global bAnzeigenGitternetzlinien         As Boolean     'ActiveWindow.DisplayGridlines
                                                        'Gitternetzlinien anzeigen.
Global lExcelFensterGroesse              As Long        'ActiveWindow.WindowState
                                                        'Excel (Applikation) Fenstergroesse.
Global iTabellenblattZoom                As Integer     'ActiveWindow.Zoom

Global bFluchttaste                      As Boolean     'Fluchttaste (Escape) wurde gedrueckt.

'-----------------------------------------------------------------------------/
'-  Globale Konstanten und Variablen - fuer das aktuelle Spiel benoetigt      -/
'-----------------------------------------------------------------------------/
Global Const MAX_ZEILEN = 6                             'Maximale Anzahl Zeilen.
Global Const MAX_SPALTEN = 8                            'Maximale Anzahl Spalten.
Global Const MAX_PAARE = MAX_ZEILEN * MAX_SPALTEN / 2 'Maximale Anzahl der Paare.

Global bZug1                 As Boolean                 'Zug 1 wurde bereits gemacht.
Global iZeileZug1            As Integer                 'Zug 1 war in dieser Zeile.
Global iSpalteZug1           As Integer                 'Zug 1 war in dieser Spalte.
Global stInhaltZug1          As String                  'Zug 1 hatte diesen Inhalt.
```

```
Global iPunkte              As Integer              'Aktuelle Anzahl der Punkte.
Global iPaare               As Integer              'Aktuelle Anzahl der Paare.
Global iFehlversuche        As Integer              'Anzahl der Fehlversuche.
Global iHighscore           As Integer              'Aktueller Highscore.
Global iNeuerHighscore      As Boolean              'Neuer Highscore erreicht.

'-------------------------------------------------------------------------/
'-  Definitionen meiner Farbwerte fuer Word und Excel ab Version 2007      -/
'-------------------------------------------------------------------------/
Global Const WEISS = 16777215               'Farbe weiss = RGB(255, 255, 255); ColorIndex = 2
Global Const ROT = 255                      'Farbe rot = RGB(255, 0, 0); Excel ColorIndex = 3
```

4.3.5.) Programmcode aus Modul Makros

Hier gleich der vollständige Programmcode. Er wurde vollständig neu erstellt bzw. aus dem Entwurf einkopiert.

```
'*************************************************************************/
'*                                                                       */
'*      PROGRAMM:       Memory                                           */
'*      PROJEKT:        Spiele programmieren in Excel                    */
'*                                                                       */
'*      MODULNAME:      Makros                                           */
'*      ARCHIVIERT:     -                                                */
'*                                                                       */
'*      VERSION:        V  01.00                                         */
'*      VERSIONSDATUM:  11 Jan 2019 12:00:00                             */
'*                                                                       */
'*      BEARBEITER:     Ing. Harald Mitsch, TBM                          */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      (C) 2019 by TBM-Technisches Buero Mitsch   Elektrotechn. Planungen  */
'*      Lizenz: FREEWARE                           und ET - Installationen  */
'*      Josefgasse 6, 3380 Poechlarn               EDV - Selfmade Software  */
'*      Tel.:02757 / 46 56 bzw. 0676 /588 09 16    allgemeines Zeichenbüro  */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      BESCHREIBUNG:   Behandelt Funktionsaufrufe in gesamter Excel-Datei. */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      INTERNE FUNKTIONEN:       Spiel_Starten                          */
'*                                                                       */
'*=======================================================================*/
'*                                                                       */
'*      AENDERUNGEN: - Modification History:                             */
'*                                                                       */
'*      V 01.00     11.01.2019    Mitsch Harald, TBM, Initial Version    */
'*                                                                       */
'*************************************************************************/
Option Explicit                                 'Alle Variablen muessen
                                                'explizit angegeben werden.
'Option Private Module                           'Die einzelnen Subs nicht unter
                                                'dem Menuepunkt Makros anzeigen.

Public Sub Spiel_Starten()
'*************************************************************************/
'*  Makro wird ausgefuehrt bei Klick auf Schaltflaeche: starten          */
'*************************************************************************/
  Call SpielStarten                             'Funktion aufrufen.
End Sub                                         'Prozedurende.
```

4.3.6.) Programmcode aus Modul Funktionen

Hier gleich der vollständige Programmcode. Er wurde vollständig neu erstellt bzw. aus dem Entwurf einkopiert. Die zuvor erwähnte Funktion **SpielfeldZeichnen** ist hier nicht mehr enthalten.

```
'*********************************************************************************/
'*                                                                               */
'*                                  \\\\|////                                    */
'*                                  \\ _ _ //                                    */
'*                                   ( o o )                                     */
'*================================oOOo-(_)-oOOo==================================*/
'* WICHTIG: Dieser Code benoetigt zusaetzliche Verweise auf die ActiveX-dll      */
'* Klick: Extras > Verweise > Visual Basic for Applications                      */
'* Klick: Extras > Verweise > Microsoft Excel 12.0 Objects Library               */
'* Klick: Extras > Verweise > Microsoft Office 12.0 Objects Library              */
'* Klick: Extras > Verweise > OLE Automation                                     */
'*=======================================Oooo====================================*/
'*                                 oooO  (  )                                    */
'*                                 (  )  ) /                                     */
'*                                  \ ( (_/                                      */
'*                                   \_)                                         */
'*                                                                               */
'*********************************************************************************/
'*********************************************************************************/
'*                                                                               */
'*      PROGRAMM:       Memory                                                   */
'*      PROJEKT:        Spiele programmieren in Excel                            */
'*                                                                               */
'*      MODULNAME:      Funktionen                                               */
'*      ARCHIVIERT:     -                                                        */
'*                                                                               */
'*      VERSION:        V  01.00                                                 */
'*      VERSIONSDATUM:  11 Jan 2019 12:00:00                                     */
'*                                                                               */
'*      BEARBEITER:     Ing. Harald Mitsch, TBM                                  */
'*                                                                               */
'*===============================================================================*/
'*                                                                               */
'*      (C) 2019 by TBM-Technisches Buero Mitsch   Elektrotechn. Planungen       */
'*      Lizenz: FREEWARE                           und ET - Installationen       */
'*      Josefgasse 6, 3380 Poechlarn               EDV - Selfmade Software       */
'*      Tel.:02757 / 46 56 bzw. 0676 /588 09 16    allgemeines Zeichenbüro       */
'*                                                                               */
'*===============================================================================*/
'*                                                                               */
'*      BESCHREIBUNG:   Behandelt Funktionen zur gesamten Excel-Datei.           */
'*                                                                               */
'*      NEBENEFFEKTE:   -                                                        */
'*                                                                               */
'*      GRENZEN:        -                                                        */
'*                                                                               */
'*===============================================================================*/
'*                                                                               */
'*      DEKLARIERTE FUNKTIONEN:   keine                                          */
'*                                                                               */
'*      EXPORTIERTE FUNKTIONEN:   SpielStarten                                   */
'*                                                                               */
'*      IMPORTIERTE FUNKTIONEN:   keine                                          */
'*                                                                               */
'*===============================================================================*/
'*                                                                               */
'*      EXPORTIERTE VARIABLEN:    keine                                          */
'*                                                                               */
'*      IMPORTIERTE VARIABLEN:    alles aus Modul Globales                       */
'*                                                                               */
'*===============================================================================*/
```

```
'*                                                                            */
'*      INTERNE FUNKTIONEN:       SpielStarten                                */
'*                                Fluchttaste                                 */
'*                                SpielfeldZeichnen                           */
'*                                                                            */
'*============================================================================*/
'*                                                                            */
'*      AENDERUNGEN: - Modification History:                                  */
'*                                                                            */
'*      V 01.00     11.01.2019    Mitsch Harald, TBM, Initial Version         */
'*                                                                            */
'******************************************************************************/
Option Explicit                                 'Alle Variablen muessen
                                                'explizit angegeben werden.

Function SpielStarten()                         'Funktion zum Starten:
'******************************************************************************/
'*      INTERNES UNTERPROGRAMM - INTERNE FUNKTION                             */
'******************************************************************************/
'*----------------------------------------------------------------------------*/
'*                                                                            */
'*      FUNKTION:       SpielStarten                                          */
'*                                                                            */
'*      BESCHREIBUNG:   Wurde die globale Konstante SPIELSTART_NEU im Modul */
'*                      Globales auf JA gesetzt, dann wird beim Oeffnen der */
'*                      Datei ein neues gemischtes Memory angezeigt. Bei    */
'*                      NEIN kann der Spieler weitermachen wo er aufgehoert */
'*                      hat. Oder man koennte das Spiel neu starten lassen. */
'*                                                                            */
'*      RETURN          Boolean ..... True = Funktion in Ordnung              */
'*                                    False = Fehler in Funktion              */
'*      PARAMETER:      keine                                                 */
'*                                                                            */
'*----------------------------------------------------------------------------*/
Dim i                   As Integer              'Allgemeine Zaehlvariable.
Dim iZufall             As Integer              'Eine Zufallszahl.
Dim iZeile              As Integer              'Aktuelle Zeilennummer.
Dim iSpalte             As Integer              'Aktuelle Spaltennummer.
Dim stKarten            As String               'Zeichen fuer Karteninhalte
Dim stKartenInhalt(MAX_PAARE * 2) As String     'Karteninhalte wie aufgelegt.
Dim stZeichen(61) As String                     'Zeichen aus Wingdings.

On Error GoTo Fehler                            'Bei Fehler zur Fehlerbehandlung.
SpielStarten = True                             'Returnwert = alles OK vordefinieren.
bFluchttaste = False                            'ESC wurde nicht gedrueckt.

Set oTabellenblatt = Sheets(TABELLENBLATT)      'Akt. Tabellenblatt festlegen.
bZug1 = False                                   'Erste Karte nicht aufgedeckt.
iFehlversuche = 0                               'Anzahl Fehlversuche ruecksetzen.
iPunkte = 0                                     'Punktestand ist Null.
iPaare = 0                                      'Punktestand ist Null.
Range("PUNKTE") = "Punkte: 0000"                'Punktestand ausgeben.
Range("PAARE") = "Paare: 00/" & Trim(Str(MAX_PAARE)) 'Aufgedeckte Paare ausgeben.
iHighscore = Range("HIGHSCORE")                 'Aktuellen Highscore auslesen.
iNeuerHighscore = False                         'Neuer Highscore nicht erreicht.
Range("I3").Select                              'Cursor auf Zelle I3 setzen.

For i = 33 To 93                                'Wingdings: brauchbare Zeichen.
  stZeichen(i - 32) = Chr(i)                    'Karteninhalte definieren.
Next i

For i = 1 To 61                                 'Zeichen aus Wingdings mischen.
  iZufall = Int(Rnd() * 61) + 1                 'Jedes Zeichen zumindest einmal
  stZeichen(0) = stZeichen(iZufall)             'vertauschen.
  stZeichen(iZufall) = stZeichen(i)
  stZeichen(i) = stZeichen(0)
Next i
```

```
stKarten = ""                                          'Verwendete Zeichen loeschen.
For i = 1 To MAX_PAARE                                 'Die ersten zufaelligen
  stKarten = stKarten & stZeichen(i)                   'Zeichen als Karteninhalt
Next i                                                 'festlegen

stKarten = stKarten & stKarten                         'Karten verdoppeln.
For i = 1 To Len(stKarten)                             'Alle Karteninhalte zum
  stKartenInhalt(i) = Mid(stKarten, i, 1)              'Mischen vorbereiten bzw.
Next i                                                 'ins Kartenfeld einlesen.

For i = 1 To MAX_PAARE * 2                             'Vorbereitetes Kartenfeld mischen.
  iZufall = Int(Rnd() * MAX_PAARE * 2) + 1             'Jede Karte zumindest einmal
  stKartenInhalt(0) = stKartenInhalt(iZufall)          'vertauschen.
  stKartenInhalt(iZufall) = stKartenInhalt(i)
  stKartenInhalt(i) = stKartenInhalt(0)
Next i

i = 1                                                  'Zaehlvariable ruecksetzen.
For iZeile = 1 To MAX_ZEILEN                           'Der Reihe die Zellen mit den
  For iSpalte = 1 To MAX_SPALTEN                       'Inhalt der Karten befuellen:
    oTabellenblatt.Cells(iZeile, iSpalte).Value = "'" & stKartenInhalt(i): i = i + 1
    oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color = ROT 'Hintergrundfarbe ROT
    oTabellenblatt.Cells(iZeile, iSpalte).Font.Color = ROT 'und Schriftfarbe ROT
  Next iSpalte
Next iZeile
                                                       'Fehlerbehandlungsroutine:
Beenden:                                               '=========================
  Application.Cursor = xlDefault                       'Sanduhr ausblenden.
  Exit Function                                        'Funktion abbrechen.

Fehler:                                                'Fehlermarke - Fehlerbehandlung:
  MsgBox Err.Description                               'Fehlermeldung ausgeben.
  Resume Beenden                                       'Bei Funktionsende weitermachen.
  Resume                                               'Nur fuer Testzwecke hinterlegt.
End Function                                           'Funktionsende.
```

4.3.7.) Programmcode aus Modul Standard

In dieses Modul wurde nur die neue Funktion **Fluchttaste** einkopiert.

```
Public Function Fluchttaste()
'*************************************************************************/
'*      INTERNES UNTERPROGRAMM - INTERNE FUNKTION                        */
'*************************************************************************/
'*----------------------------------------------------------------------*/
'*                                                                      */
'*      FUNKTION:       Fluchttaste                                     */
'*                                                                      */
'*      BESCHREIBUNG:   Schliesst die aktuelle Excel-Datei sofort und ohne  */
'*                      zu speichern. Zuvor werden noch alle Hintergrund-   */
'*                      prozesse beendet und die Entwickleransicht wird     */
'*                      wieder hergestellt. Excel selbst bleibt geoeffnet.  */
'*                                                                      */
'*      RETURN          nichts                                          */
'*                                                                      */
'*      PARAMETER:      keine                                           */
'*                                                                      */
'*----------------------------------------------------------------------*/
On Error Resume Next                                   'Bei Fehler einfach weitermachen.
bFluchttaste = True                                    'ESC wurde jetzt gedrueckt.
Call SpielAnsichtAktualisieren(NEIN)                   'Umschalten auf Entwickleransicht.

ActiveWorkbook.Close savechanges:=False                'Excel-Datei sofort schliessen.
End Function                                           'Funktionsende.
```

4.3.8.) Schummeln verhindern

Vielleicht haben Sie es schon bemerkt. Wenn wir mehrere verdeckte Karten mit der Maustaste selektieren, dann kann man erkennen, welche Motive sich hinter den einzelnen Karten verbergen. Das wollen wir natürlich verhindern.

Dazu dürfen wir das Symbol nicht in der Zelle speichern sondern in einem Feld. Erst wenn man eine Karte anklickt soll das Motiv in die Zelle geschrieben und angezeigt werden. Wurde das Paar gefunden, dann bleiben die Symbole eingeblendet, ansonsten wird die Zelle wieder geleert.

Zunächst legen wir und im Modul **Globales** eine neu globale Variable an:

```
Global iKarteninhalt(MAX_ZEILEN, MAX_SPALTEN) As Integer 'Ascii-Code wie aufgelegt.
```

Diese befüllen wir dann in der Funktion **SpielStarten** im Modul **Funktionen**. Dazu setzen wir die Programmzeile mit der Zuordnung des Karteninhaltes unter Kommentar. Stattdessen schreiben wir den Ascii-Code des Karteninhaltes in unsere neue globale Variable. In die Zelle selbst schreiben wir einen Leerstring.

```
For iZeile = 1 To MAX_ZEILEN                          'Der Reihe die Zellen mit den
  For iSpalte = 1 To MAX_SPALTEN                      'Inhalt der Karten befuellen:
    'oTabellenblatt.Cells(iZeile, iSpalte).Value = "'" & stKartenInhalt(i): i = i + 1
    iKarteninhalt(iZeile, iSpalte) = Asc(stKartenInhalt(i)): i = i + 1
    oTabellenblatt.Cells(iZeile, iSpalte).Value = ""

    oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color = ROT 'Hintergrundfarbe ROT
    oTabellenblatt.Cells(iZeile, iSpalte).Font.Color = ROT 'und Schriftfarbe ROT
  Next iSpalte
Next iZeile
```

Im Objekt **Tabelle1 (Memory)** in der Prozedur **Worksheet_SelectionChange** lassen wir bei jedem Klick das Symbol eintragen. Stimmen beide Motive nicht überein, dann werden die eingetragenen Symbole wieder durch einen Leerstring ersetzt.

```
If iZeile > 0 And iZeile <= MAX_ZEILEN And _
   iSpalte > 0 And iSpalte <= MAX_SPALTEN And _
   oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color <> WEISS Then
   oTabellenblatt.Cells(iZeile, iSpalte).Value = "'" & Chr(iKarteninhalt(iZeile, iSpalte))

  If bZug1 = False Then                               'Erste Karte nicht aufgedeckt:
    oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color = WEISS '1. Karte anzeigen.
    bZug1 = True                                      'Merken 1. Karte ist aufgedeckt.
    iZeileZug1 = iZeile                               'Merken Zeile der Karte.
    iSpalteZug1 = iSpalte                             'Merken Spalte der Karte
    stInhaltZug1 = oTabellenblatt.Cells(iZeile, iSpalte).Value 'und Karteninhalt.
  Else                                                '2. Karte wird umgedreht:
    oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color = WEISS '2. Karte anzeigen.
    'Wenn Inhalt der 1. Karte ungleich der 2. Karte, dann etwas warten und dann
    If stInhaltZug1 <> oTabellenblatt.Cells(iZeile, iSpalte).Value Then
      Sleep (750)                                     'beide Karten unsichtbar machen:
      oTabellenblatt.Cells(iZeileZug1, iSpalteZug1).Interior.Color = ROT
      oTabellenblatt.Cells(iZeile, iSpalte).Interior.Color = ROT
      oTabellenblatt.Cells(iZeileZug1, iSpalteZug1).Value = ""
      oTabellenblatt.Cells(iZeile, iSpalte).Value = ""
      ...
```

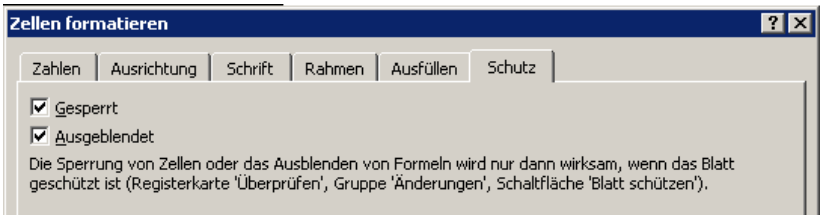
Damit haben wir dem Spieler die Möglichkeit genommen unter die Karten zu schauen.

5.) Schlußbemerkungen

Für ein Spiel sollten Sie auch ein **Icon** erstellen, welches man einer Verknüpfung zuweisen kann. Zum Erstellen eigener Icons verwende ich den kostenlosen **Greenfish Icon Editor Pro**. Damit wurde auch das Icon für das Memory-Spiel erstellt:

Bei komplexeren Spielen sollte auch eine detaillierte **Spielbeschreibung** erstellt werden.

Definieren Sie den Schutz für die Zellen im Tabellenblatt. Am besten zuerst alle Zellen markieren und dann den Schutz für alle Zellen aktivieren:



Danach jene Zellen auswählen, welche vom Programm beschrieben werden, hier die Zellen I4 bis I7 (Highscore, Punkte, Paare, Ausgabe Highscore) sowie alle Zellen welche die Karten repräsentieren und dort das Häkchen bei "Gesperrt" entfernen.

Aktivieren Sie den Blattschutz in Excel. Im Register **Überprüfen**, Symbol **Blatt schützen**:

Bei Memory muß zusätzlich die Berechtigung: **Zellen formatieren** (da wir die Hintergrundfarbe der Karten verändern) abgehakt werden.

Schützen Sie zusätzlich Ihr **VBA-Projekt** mit einem Paßwort wenn Sie den Programmcode geheim halten wollen. In Visual Basic das VBA-Projekt mit der rechten Maustaste anklicken -> Eigenschaften von VBA-Projekt… -> Registerblatt: Schutz. Dort zweimal das Kennwort eingeben und Klick auf OK.

So das wär´s dann wieder Mal – viel Spaß beim Nachprogrammieren oder Überarbeiten des Spieles.

Noch ein letzter Hinweis: Sollte nach dem Starten die Zelle A1 nicht rot ausgefüllt sein, dann ist zusätzlich in der Funktion **SpielAnsichtAktualisieren** im Modul **Standard** die Zeile:

```
'Range("A1").Select                    'Zelle A1 selektieren.
Range("I3").Select                     'Cursor auf Zelle I3 setzen.
```

unter Kommentar zu setzen. Anstelle von A1 springe ich standardmäßig die gesperrte Zelle I3.