

```

{ AuD Seminar 3 - Aufgabe 4. }

{ Schreiben Sie ein Pascal-Programm, das 10 Integers einliest }
{ und den Komponenten einer geeigneten Reihung (Array) zuweist. }
{ Diese Zuweisung soll auf jeden Fall stattfinden! Die kleinste }
{ und größte Zahl in der Reihung sind zu bestimmen und auszugeben. }

{$X+} { Extended Syntax }
Uses Crt;

Var
  a : Array[0..9] Of Integer; { Array }
  s : String; { Dummy-String }
  Err : Integer; { Errorvar. für Val }
  i : Byte; { Zählvar. für For }
  Min : Array[Boolean] Of Integer; { Kleinste Zahl von a }
  Max : Array[Boolean] Of Integer; { Größte Zahl von a }

Begin
  ClrScr;
  For i := 0 To 9 Do Begin
    Repeat
      Write ('Bitte geben Sie die ', Chr(i + 48), '. Zahl ein: ');
      ReadLn (s);
      If s = '' Then s := '0'; { Fehlerbehandlung }
      Val (s, a[i], Err); { Umwandlung von s in Int-Wert }
    Until Err = 0;
    If i = 0 Then Begin
      Min[False] := a[0]; Min[True] := 0;
      Max[False] := a[0]; Max[True] := 0;
    End
    Else Begin
      If a[i] < Min[False] Then Begin
        Min[False] := a[i];
        Min[True] := i;
      End;
      If a[i] > Max[False] Then Begin
        Max[False] := a[i];
        Max[True] := i;
      End;
    End;
  End;
  WriteLn (#10#13, 'Min: ', Min[False]:6, ' (', Min[True], ')',
    #10#13, 'Max: ', Max[False]:6, ' (', Max[True], ')');
  ReadKey;
End.

```

```

{ AuD Seminar 3 - Aufgabe 5. }

{ Schreiben Sie ein Pascal-Programm, das zwei Matrizen des Formates 5 x 3 }
{ addiert. Matrizen werden addiert, indem die entsprechenden Elemente }
{ addiert werden: }
{ c[i, j] = a[i, j] + b[i, j] für i = 1(1)5, j = 1(1)3 }

{$X+} { Extended Syntax }

Uses Crt;

Type
  TType = Byte;

Var
  m1, m2, m3 : Array[0..4, 0..2] Of TType;
  i, j       : Byte;

Begin
  ClrScr; Randomize;
  GoToXY (01, 2); Write ('m1 = ');
  GoToXY (28, 2); Write ('m2 = ');
  GoToXY (53, 2); Write ('m1 + m2 = ');
  For j := 0 To 2 Do Begin
    For i := 0 To 4 Do Begin
      m1[i, j] := Random(50); { mgl. Wertzuweisung von m1 }
      m2[i, j] := Random(50); { mgl. Wertzuweisung von m2 }
      m3[i, j] := m1[i, j] + m2[i, j];
      GoToXY (3 * (i + 1) + 3, j + 1); Write (m1[i, j]:2);
      GoToXY (3 * (i + 1) + 30, j + 1); Write (m2[i, j]:2);
      GoToXY (3 * (i + 1) + 60, j + 1); Write (m3[i, j]:2);
    End; WriteLn;
  End;
  ReadKey;
End.

```

```

{ AuD Seminar 3 - Aufgabe 6. }

{ Eine quadratische Matrix vom Format 21 x 21 soll mit Integer- }
{ Zufallszahlen, die zwischen 0 und 10 liegen, zeilenweise gefüllt }
{ werden. Eine Zufallszahl kann mit der Standardfunktion random(10) }
{ erzeugt werden. Anschließend sollen die Zahlen, welche auf den im }
{ Bild mit fetten Strichen markierten Positionen stehen, summiert }
{ werden. (In Schnittpunkten der Linien stehende Zahlen sind nur }
{ einmal zu addieren! Die Fetten Linien sollen die senkrechten Linien }
{ und die Diagonalen sein.) Verwenden Sie nur minimal notwendige }
{ Laufanweisungen (Zählschleifen). }

{$X+} { Extended Syntax }

Uses Crt;

Var
  Matrix : Array[0..20, 0..20] Of Byte; { Matrix 21 x 21 Felder }
  i, j : Byte; { For-Zählvar. }
  Sum : Word; { Summe der Diagonalen + Seiten }

Begin
  ClrScr; Randomize;
  Sum := 0;
  WriteLn;
  For i := 0 To 20 Do Begin
    For j := 0 To 20 Do Begin
      Matrix[i, j] := Random(10);
      If (j = 0) Or (j = 20) Or (i = j) Or (20 - j = i) Then Begin
        TextColor (10);
        Inc (Sum, Matrix[i, j]); { Sum := Sum + Matrix[i, j] }
      End
      Else TextColor (7);
      Write (Matrix[i, j]:2);
    End;
    WriteLn;
  End;
  WriteLn (#10, ' Summe der markierten Linien: ', Sum);
  ReadKey;
End.

```