Chapter II/SECTION 1/2/3

```
bld020101.m  Interpolation of 1/(1 + x*x)
bld020102.m  Lagrange polynomials for n = 3
bld020103.m  Example for Bezier polynomial of degree n = 3
bld020104.m  Hermite polynomials for n = 3 , H_0i
bld020105.m  Hermite polynomials for n = 3, H_1i
bld020106.m  Bezier curve for Bezier polynomials of degree n = 3
bld020107.m  Spline curve
bld020201.m  Legendre polynomials
bld020202.m  Chebyshev polynomials for n = 3
demo1.m      Test of integration rules for a triangle
```

Chapter II/SECTION 4: Initial Value Problems

```
bld020401.m   Exact solution of DGL
bld020402.m   Differential equation, Euler explizit
bld020403.m   Differential equatin, trapezoidal rule
bld020405.m   Program for plots of the stability regions
              of one-step methods
bld020406.m   Stability region of Rosenbrock's method
bld020407a.m  Stability region of explicit Adams' methods
bld020407b.m  Stability regions of implicit Adams' methods
bld020408.m   Stability regions of multistep methods
demo1.m       Arenstorf orbits by using dopri.m,
dopri.m       MATLAB version of FORTRAN version of HAIRER I
```

Chapter II/SECTION 5: Boundary Value Problems

```
adapt01.m    Adaption of shooting points
box.m        Box scheme for Newton's method
bsp01.m      Example of Stoer-Bulirsch, Par. 7.3, Bsp.01
demo1.m      Masterfile for multiple shooting methods
mehrziel.m   Multiple shooting scheme for Newton's method
newton.m     Quasi-global Newton's method
```

Chapter II/SECTION 6: Periodic Problems

```
bsp01.m      Nerve membran model
bsp02.m      Heat flow problem
bsp03.m      Arenstorf orbit I
bsp03.tex    Calculations to BSP03.M
demo1.m      Masterfile for multiple shooting method
mehrziel_p.m Multiple shooting scheme for Newton's method
             and problems with unknown period
newton_p.m   Quasi-global Newton's method for periodic problems
```

Chapter III/SECTION 1/2/3/4, Linear-Quadratic Programming

```
bfgs.m      BFGS method
demo1.m     Example, bfgs.m and desc.m
demo2.m     Test of dlqp.m
demo3.m     Test of dlqp.m  with random variables
desc.m      Steepest descend
```

dlqp.m   Linear-quadratic Programming after Goldfarb-Idnani
dlqp_g.m   as dlqp.m, bur only inequalities
ga_test.m   Goldstein-Armijo descend test

Chapter III/SECTION 5, Nonlinear Programming
bsp01.m--bsp16.m Examples
demo1.m   Masterfile for gradient projection
demo2.m   Masterfile for sequential quadratic programmiing
gp.m   Gradient projection method general
gp_g.m   Gradient projection method, only inequalities
restor.m   Restoration in gp.m
sigini.m   Start vector for step length sigma in gp.m
sqp.m   Sequential quadratic programming general
sqp_g.m   Sequential quadratic programming, only inequalities

Chapter III/FEXIPLEX, Method of Nelder-Mead
demo1.m   Minimization of a function (3 Ex.)
demo2.m   Minimization with constraints (4 Ex.)
simplex.m Minimization after Nelder and Mead

Chapter IV/CONTROL01, Control Problems
Solution by the method sqp.m of Chapter/SECTION_5
demo1.m Masterfile with sqp.m, examples 1--9
demo3.m Reentry problem, Stoer, p. 491, US units, SI units
demo4.m Space craft X-38 without constraints
demo5.m Space craft X-38 with constraints of sign of
         attacking angle GAMMA

Chapter IV/CONTROL02, Control Problem transformed into
          Boundary value problem
box.m   Box scheme for NEWTON's method
bsp01.m   Thrust problem, control eliminated
bsp02.m   Orbit problem, control eliminated
bsp03.m   Zermelo's problem, costate eliminated
demo.m   Masterfile for NEWTON's method
newton.m Globalised NEWTON's method

Chapter IV/CONTROL03, Control Problem and Gradient Method
demo1.m   Simple example after Dyer-McReynolds, p. 127
demo2.m   Brachistochrone, Dyer-McReynolds, p. 128
demo3.m   Orbit problem, Bryson-Ho, p.66, Dyer-McReynolds, p.73
demo4a.m Thrust problem Bryson-Ho, par. 2.4, Start trajectory
demo4b.m Thrust problem Bryson-Ho, par. 2.4, solution
grad01.m -- grad04.m Gradient method

Chapter V/SECTION 6, HOPF Bifurcation
conjgrad.m   Method of conjugate gradients after Stoer
cg_lq.m   Method of conjugate gradients after Allgower/Georg

```
demo1.m      Masterfile for HOPF bifurcation with backward
             differentiation or with trig. collocation
demo2.m      Masterfile for continuation in DEMO1.m only for
             backward differentiation
hopf_bdf.m   Hopf bifurcation with backward differentiation
hopf_trig.m  Hopf bifurcation with trig. collocation
hopf_contin  Simple continuation after HOPF.M
```

Chapter V/SECTION 7, Numerical Bifurcation

```
demo1.m   Pitchfork bifurkation
demo2.m   Example of Crandall (4 branching points)
demo3.m   Poisson's equation in unit square
          Newton's method (6 examples)
demo4.m   Poisson's equation in unit square
          direct iteration (6 examples)
demo5.m   Continuation by MU for Poisson's equation
bif.m     Direct iteration method
```

Chapter V/SECTION 8, Continuation Method

```
demo1.m   Masterfile for continuation after Allgower/Georg
demo2.m   Masterfile for continuation after Rheinboldt
cont.m    Continuation after Allgoweer/Georg
pitcon1.m -- picon5.m  Continuation after Rheinboldt
newton.m  Newton's method for PITCON.M
```

Chapter VI/SECTION_2_3_4, Central Fields

```
demo1.m    Graphics for Kepler's second law
demo2.m    Motion in central field, different potentials
demo3.m    Arbitrary conic sections under different
           initial positions and velocities
kepler.m   Computes conic section by intial data
ellipse.m  Draws ellipse with data
parabel.m  Draws parabola with data
hyperbel.m Draws hyperbola with data
```

Chapter VI/SECTION_5, Three-Body Problem

```
arenstorf.m Different Arenstorf orbits
demo1.m Two-body problem, trajectories by differential system
demo2.m Three-body problem, trajectories by differential system
```

Kapitel VI/SECTION_6_7, Top

```
demo1.m  Computes EULER angles for top and trajectory
         of top's axis directly by EULER-LAGRANGE equations
demo2.m  Top demo, the 7 examples
demo3.m  Computes EULER angles phi and theta by initial data of
         DEMO2.M with differential system and trajectory of
         top's axis
demo4.m  Draws curve of Euler angle theta and curve of
         derivation of phi
demo5.m  Movie for top
```

```
Chapter VII/SECTION 3 beam in special position
demo4.m         Masterfile, bending beam
balkelement1.m  Beam element
balken1.m       Beam in special position
balken2.m       Beam in general plane position
```

```
Chapter VII/SECTION 4, Frameworks of rods
demo1.m        Masterfile, forces in plane framework
demo2.m        Masterfile, displacements in plane framework
               with image sequence
demo3.m        Masterfile, displacements in spatial framework
stabelement1.m Tension rod in plane position
stabelement2.m Tension rod in spatial position
stabwerk1.m    Forces in plane framework
stabwerk2.m    Displacements in plane framework
stabwerk3.m    Displacements in spatial framework
```

```
Chapter VII/SECTION 5, spatial frameworks
demo1.m     Masterfile for spatial frameworks
balken2.m   Beam element, nearly general position
rahmen2.m   Displacements in spatial frameworks
```

```
Chapter IX/FEM_1, Elliptic boundary value problems
demo1.m       Example, linear triangular elements
demo2.m       Example, lineare parallelogram elements
demo3.m       Example, quadratic triangular elements
demo4.m       Example, quadratic triangular and parallelogram
              elements
demo5.m       Example, cubic triangular and parallelogram
              elements
demo6.m       Example, isopar. quadratic triangular and
              quadrilateral elements
ellipt1.m     Linear  triangular element
ellipt2a.m    Linear  parallelogram element
ellipt2b.m    Isopar. quadrilateral element
ellipt3.m     Quadratic triangular and parallelogram element
ellipt4.m     Cubic triangular and parallelogram element
ellipt5.m     Isopar. triangular and quadrilateral element
fem_bilin.m   Bilinear parallelogram element
fem_drlell.m  Linear triangular element
fem_drkell.m  Cubic triangular element after Zienkiewicz
fem_drqell.m  Quadratic triangular element
fem_isobil.m  Isopar. bilinear quadrilateral element
fem_isodrq.m  Isopar. quadratic triangular element
fem_isopaq.m  Isopar. quadratic quadrilateral element,
              Serendipity class
fem_isoraq.m  Isopar. quadratic boundary element
fem_pakell.m  Cubic parallogram element, Serendipity class
```

```
fem_rakell.m  Cubic hermitean Boundary element
fem_raqell.m  Quadratic boundary element
fem_ralell.m  Linear boundary element
fem_ffqdre.m  Shape functions for FEM_ISODRQ.M
fem_ffqbil.m  Shape functions for FEM_ISOBIL.M
fem_ffqpas.m  Shape functions for FEM_ISOPAQ.M
fem_ffquad.m  Shape functions for FEM_ISORAQ.M
myadapt.m     Simple adaptive mesh refinement

Chapter IX/FEM_2, Discs and plates
bsp021g.m     Spanner, geometry data
bsp021h.m     Spanner, boundary data, loads
bsp022.m      Nine examples for plates
fem_batoz.m   Non-conforming quadratic triangular element
fem_batoz1.m  Auxiliary file for FEM_BATOZ.M
```

```
fem_drkpla.m  Non-conforming cubic triangular element
              after ZIENKIEWICZ
fem_drksch.m  Cubic disc element with condensation
              in triangle
fem_elstif.m  Non-conforming quadratic triangular element,
              other version
fem_pakpla.m  Non-conforming quadratic parallelogram element
              Serendipity class
fem_ripla.m   Conforming bicubic rectangular element
demo1.m       Masterfile for disc problems
demo2.m       Masterfile for plate problems after H.R.SCHWARZ
demo3.m       Masterfile for plate problems after BATOZ
scheibe3.m    Disc problem, cubic triangular element
spaqua1.m     Stress computation for cubic triangular element

Chapter IX/FEM_3, Navier-Stokes Equations
Stream-function vorticity form
Time-dependent form after H.Ninomiya/K.Onishi; artificial
boundary conditions for vorticity automatically generated .
Time-independent form as elliptic system after Barragy-Carey.
demo1.m      lid driven cavity, time-dependent
demo2.m      flow past half cylinder, time-dependent,
demo3.m      flow past cylinder, time-dependent,
demo4.m      backfacing step, time-dependent
demo5.m      NS-part for transport problem, time-dependent
demo6.m      Example with exact solution, time-dependent
demo7.m      lid driven cavity, time-independent,
             Simple iteration
demo8.m      Example with exact solution, time-independent,
             Simple iteration
demo9.m      Example with exact solution, time-independent,
             Newton's method
demo10.m     Example with exact solution, time-dependent,
             with ode23.m
ellipt1.m:   Computes stream function by Poisson's equation
prepar.m     Mesh generation (with PDE TOOLBOX)
rside10.m    Right side for differential equation in ode23.m
velocity.m   Computes flow by stream function
vorticity.m  Computes vorticity
wbound.m     Computation of artificial boundary conditions
             for vorticity
```

```
Chapter IX/FEM_4, Convection
Stream-function vorticity form
Time-dependent form after H.Ninomiya/K.Onishi; artificial
boundary conditions for vorticity automatically generated .
Time-independent form as elliptic system after W.N. Stevens
demo1.m        Thermal flow in a cup, time-dependent
demo2.m        Convection in a closed compartment, time-dependent
demo3.m        Convection in a square box, time-dependent
demo4.m        Thermal flow in a cup, time-independent
demo5.m        Convection in a unit square, time-independent
demo6.m        Example with exact solution, time-independent
convection.m   Computes temperature
vorticity_k.m  Computes vorticity for convection
lanscape.m     Neumann's boundary condition
matrizen.m     Matrices for coupled system
rightsides.m   Right sides for coupled system
```

```
Chapter XI/STOKES, Navier-Stokes problems in (u_1,u_2,p)-form
Fix one value of pressure p!
demo1.m: lid driven cavity with Taylor-Hood elements
         linear: without convection term
demo2.m: lid driven cavity with Mini elements
         linear: without convection term
demo3.m: lid driven cavity with Taylor-Hood elements
         nonlinear: with convection term, simple iteration
demo4.m: unit square with Taylor-Hood elements, example with
         exact solution, linear: without convection term
demo5.m: lid driven cavity with Taylor-Hood elements
         nonlinear: with convection term, NEWTON iteration
         simple continuation possible until NU = 0.002106
         Sequel for NU: [0.1,0.05,...,0.01,0.009,...0.003,
                         0.0029,..0.0022,0.00219,...0.002106]
demo6.m.M: Letters F E M with Taylor-Hood elements
         linear: without convection term
```

```
Chapter IX/TIDAL, Shallow Water Equations
This directory contains MATLAB versions of BASIC programs
of H.Ninomiya/K.Onishi and further applications
demo1a.m     Island in a bay
demo1b.m     Island in a bay, different boundary computation
demo2.m      Finite channel with ode23.m
demo3.m      Long channel
demo4.m      Long wave on beach
flow_1.m     Velocity and water depth with lumped mass matrix
flow_2.m     As flow_1.m but with selective lumping
flow_3.m     As flow_1.m but with full mass matrix
lanscape.m   Island in a bay (geometry data, coast)
rside1.m     Right side of differential system
vnomal.m     Velocity at boundary (coast)
vnomal_n.m   Velocity at boundary (coast) (different way)
```

AAMESH: Domain Decomposition etc.
Make MATLAB-Path to AAMESH perpetual!!!

```
mesh01.m    Uniform mesh refinement, triangles, parallelograms,
            quadrilaterals
mesh01_t_q  Uniform mesh refinement simultaneous for triangles
            and parallelograms
mesh02.m    Moves node manually for triangles and quadrilaterals
            finish by pointing on the frame!
mesh03.m    Mesh smoothing for triangular decompositions of
            domains with a single boundary; long edges are
            replaced by short edges
mesh04.m    Eliminates double nodes in node matrix p and the
            field of node numbers FIELD;
mesh06_t.m  Computation of mid-points of edges for straight
            quadratic triangular elements with elimination of
            double points
mesh06_t_q.m Computation of mid-points of edges for triangles
            and quadritlaterals appearing simultaneously, with
            elimination of double points
mesh08.m    Generates equidistant nodes in a square containing
            a domain given by boundary ''e''
mesh09.m    DELAUNAY triangularization, external boundary and
            one internal boundary, in non-convex domains
            triangles exterior of domain are cancelled uses
            DELAUNAY.M of MATLAB PDE-TOOLBOX
mesh10.m    Moves nodes into center of surrounding polygon,
            runs in ascending and then in descending order
            of enumeration, for triangles and quadrilaterals
            and simply connected domains
mesh11.m    Triangularization of a domain with one possible
            cavity without interior points (also possible by
            applying DELAUNAY)
mesh12.m    Exhausts a domain without cavity by squares and the
            rest by triangles
mesh13.m    Computes mid-points and normals in mid-points of
            edges for triangles
mesh14.m    Domain decompositin by offsetting of normals,
            only simple domains
mesh15.m    Computes new row of nodes by offsetting of normals
mesh17.m    Mesh refinement for triangular decomposition of
            domains without cavity by halving of longest edge
mesh23.m    Computes numbers of triangular elements belonging
            to boundary and associated node vector, for domains
            without cavity
mesh24.m    Computes triangles with vertices contained in node
            vector U, uses PDE-TOOLBOX
mesh27.m    Eliminates triangles at the exterior of a domain
            with one cavity
```

mesh40.m	Computes neighboring nodes and adjacent triangle for boundary nodes
mesh43.m	Computes normal vector and neighboring nodes for boundary bodes
demo1.m	Demo for mesh01.m, mesh10.m, mesh11.m
demo2.m	Demo for mesh12.m
demo3.m	Demo for mesh13.m
demo4.m	Demo for mesh17.m
demo5.m	Demo for mesh14.m and mesh15.m
demo6.m	Demo for mesh02.m, mesh03.m, mesh10.m, mesh27.m
demo7.m	Demo for mesh40.m and mesh43.m
demo8.m	Demo for mesh23.m, mesh24.m

Chapter XI/SECTION 4, Dancing Discs
At first both discs are to be constructed by SCHEIBE01.M -- SCHEIBE24.M.
Both discs must touch each other at beginning.

demo1.m	Draws disc by manual input
demo2.m	Rolling of disc A onto or in disc B with DISC_ROTATE.M
demo3.m	Rolling of disc A onto or in disc B with BISECTION.M
demo4.m	Movie for discs
bisection.m	Method of bisection for computation of rotational angle
disc_aendern.m	Geometry for DEMO1.M
disc-rotate.m	Geometry for DEMO2.M