Higher Mathematica:

Modeling Differential and Difference Equations

International Compact Course

for Graduate and PhD students

University of Szeged, June 9-12, 2014

Lecturer, coordinator: János Karsai associate professor, University of Szeged, karsai.janos@math.u-szeged.hu

Length: 4x8 hours in computer cabinet

Web: www.model.u-szeged.hu (menu: Education)

Language: English

Schedule: Classes will be held in the intervals 8.30 - 12.00 and 13.00 - 16.30 with a short break in the middle.

Audience: Mathematics, Physics, Chemistry graduate and PhD student are preferred but other fields are also welcome

Prerequisites: Knowledge of Mathematica at basic level; courses of master level on differential and difference equations. Programming experience is advantageous.

Conditions:

- Participation is free, supported by the IPA HUSRB/1203/221/024 projects "Non-Standard Forms of Teaching Mathematics and Physics". Participants have to mention this support when the participation is referred.
- Participants should bring their laptops with Mathematica 9 installed. 30 days trial license is available.
- The organizers try to support the accommodation for the participants, with priority for students from the HU-SRB cross-border region.
- Travelling expenses are covered by the participants

Method: The participants and the lecturer work on computer simultaneously. In every topic, a short introduction and description are followed by solving practical problems and developing applications with *Mathematica*.

Handouts: Participants will receive the following interactive collections (More collections are available on www.model.u-szeged.hu)

[1] Mathematical and visualization packages: Mathematica, course material
[2] Computer-aided study of mathematical models with Mathematica, course material

Tentative Program

The schedule of the program below can change according to the special interest of the audience

A projekt az Európai Unió társfinanszírozásával valósul meg.
Projekat sufinansira Evropska Unija
The project is co-financed by the European Union

Day 1

- **Summary of basic concepts of the language of Mathematica**
  - Structures, types, Head, Head operations,
  - Lists in more details, Sequences
  - Value setting, rules (immediate and delayed),
  - Patterns, type check in rules and functions
  - functions vs. expressions; pure form of functions in more details
  - Piecewise or conditional definition of functions, recursions
  - Formula manipulations, logic
- **Summary of visualization:**
  - Built-in plots in 2D and 3D, dynamic visualization, graphics structures.
  - Exercise: Visualization of a moving point in 2D and 3D
- **List programming**
  - Rule-based programming
  - Structure operations on lists: Map, Apply, Thread, Fold, ...
  - Rotating lists, and applications to problems in geometry and numerical algorithm
- **Operations over functions**
  - Example: Derivative and D
  - Operations: InverseFunction, Composition, Operate, Through,..
  - Special function objects: Function, InterpolatingFunction, BooleanFunction, Transformations,....
- **Basic tools for differential equations**
  - Vector fields, streams in Mathematica
  - Symbolic and numeric solution of differential equations, interpolation
  - Elementary modeling with 1D-3D ODE's, complex case studies
  - Advanced visualization of functions and parametric curves

Day 2

- **Graphics programming structures and operations**
  - Graphics and Graphics3D, GraphicsComplex
  - How the built-in plots work
  - Applications of structure and rule-based programming to graphics objects:
  - Some advanced applications to scientific and engineering visualizations: functions, vector fields and scalar fields
- **Qualitative methods for differential equations**
  - Investigation of linear systems
  - Qualitative method 1: Stability by linearization
  - Technical interrupt: Advanced visualization of scalar fields
  - Qualitative method 2: Stability by auxiliary functions (Liapunov's second method)
  - Visualization of families of trajectories, the method of phase mapping.

Day 3

- **Iteration, nesting**
  - Recursion vs. iterations
  - Iterations, fixed points of mappings
  - Numerical applications: Newton iteration, gradient method, Euler method to solve ODE's, Picard iteration, etc,
- **Applications to difference systems**
  - Solving, visualizations
  - Visualization: Cobweb diagram
  - Fixed points, solution, stability of fixed points
  - Example: the logistic mapping with *Mathematica*, bifurcation diagram
  - Special tools in difference calculus
  - Tools for Discrete Calculus
  - Discretization of ODE's, PDE's
- **Advanced tools for differential equations**
  - Hybrid systems: WhenEvent, DiscreteVariables
  - Differential systems with Dirac delta, impulsive systems
  - Numerical solution of delay differential equations
  - Numerical solution of parametric differential equations
  - Partial differential equations

Day 4

- **Advanced applications for difference systems**
  - Program development: the Euler's method
  - Discretization of ODE's, PDE's, moving average, image processing (…) by rotating lists
  - Cellular automata
  - Iterative forms, fractal constructions: simple constructions, generating trees
- **Additional fields 2: How to develop dynamic applications**
  - Introduction to dynamic structures and operations, control objects, event handler
  - Exercises to develop lecture presentations
- **Additional fields 3: notebook development**
  - Advanced notebook operations: options, option inspector
  - Stylesheet design, automatic numbering, hyperlinks, ...
- **Summary, discussion**

A projekt az Európai Unió társfinanszírozásával valósul meg.
Projekat sufinansira Evropska Unija
The project is co-financed by the European Union