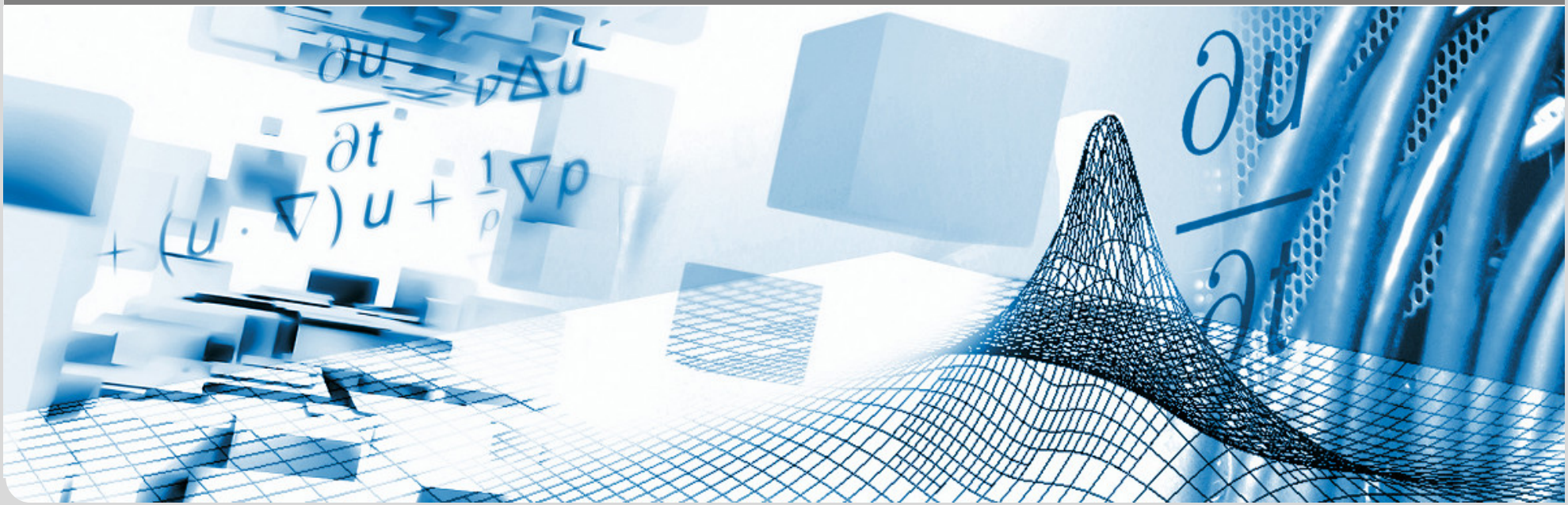
Karlsruhe Institute of Technology

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

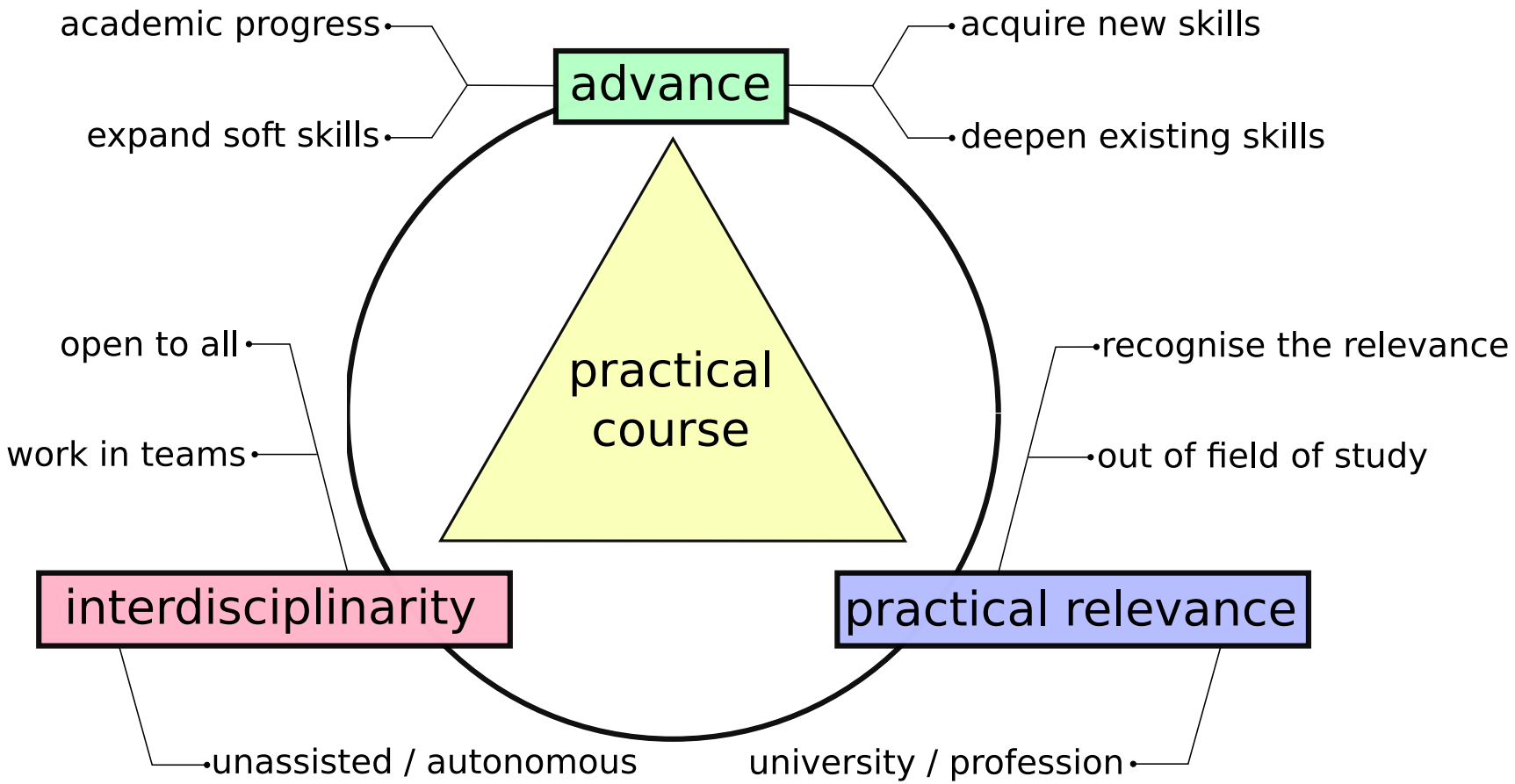
Interdisciplinary practical course on parallel finite element method using HiFlow^3

E. Treiber, S. Gawlok, M. Hoffmann, V. Heuveline, W. Karl | EuroEDUPAR, 2015/08/24

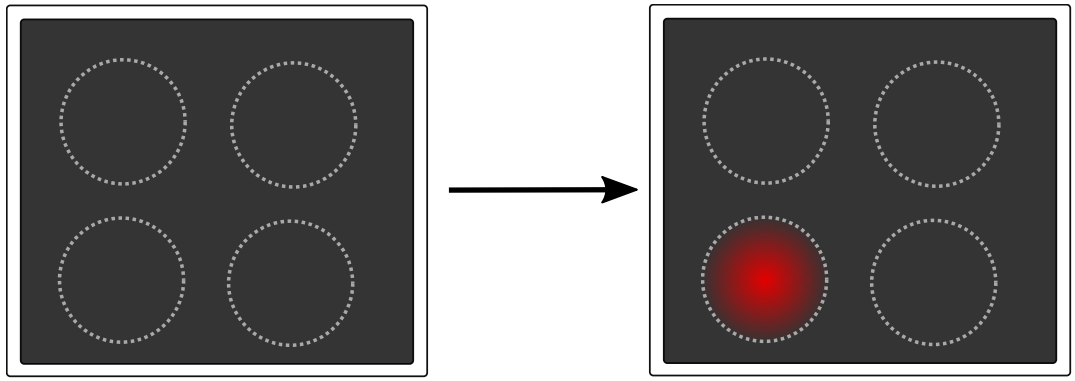
KARLSRUHE INSTITUTE OF TECHNOLOGY - ITEC/CAPP, HEIDELBERG UNIVERSITY - IWR/EMCL



Motivation


academic progress
acquire new skills
advance
expand soft skills
deepen existing skills
practical course
open to all
recognise the relevance
work in teams
out of field of study
interdisciplinarity
practical relevance
unassisted / autonomous
university / profession

Basic course content



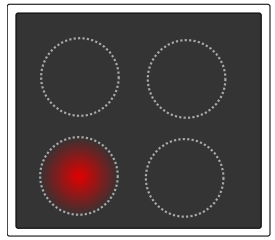
$$\frac{\partial}{\partial t} u(\vec{x}, t) - \alpha \Delta u(\vec{x}, t) = f(\vec{x}, t) \qquad \vec{x} \in \Omega, \;\; t \in (t_0, \tau)$$

$$u(\vec{x}, t) = g(\vec{x}, t) \qquad \vec{x} \in \Gamma, \;\; t \in (t_0, \tau)$$

$$u(\vec{x}, t_0) = u_0(\vec{x}) \qquad \vec{x} \in \Omega$$

Basic course content

$$\Omega \cup \Gamma = [0,1]^2 : \quad -\Delta u(\vec{x}) = f(\vec{x}) \quad \vec{x} \in \Omega$$
$$u(\vec{x}) = 0 \qquad \vec{x} \in \Gamma$$



Find $u \in H_0^1(\Omega) := \{w \in H^1(\Omega) \mid w|_\Gamma = 0\}$ such that

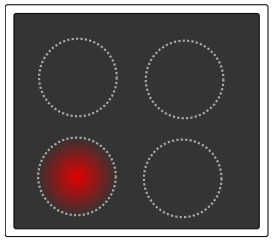
$$\underbrace{\int_\Omega \nabla u \cdot \nabla \phi \; dx}_{=:a(u,\phi)} = \underbrace{\int_\Omega f\phi \; dx}_{=:b(\phi)} \qquad \forall \phi \in H_0^1(\Omega)$$

“weak formulation“

Basic course content

Find $u \in H_0^1(\Omega)$ such that

$$a(u, \phi) = b(\phi) \qquad \forall \phi \in H_0^1(\Omega)$$



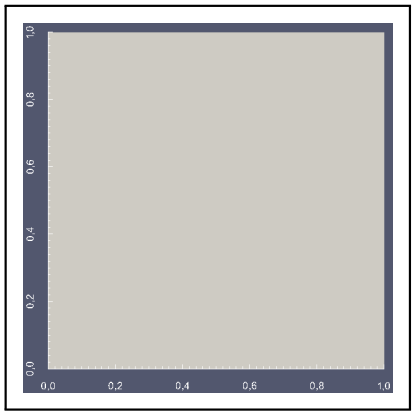
Find $u_h = \sum_{i=1}^{N} x_i \psi_i \in V_h$ such that

$$a(u_h, \phi_h) = b(\phi_h) \qquad \forall \phi_h \in V_h$$

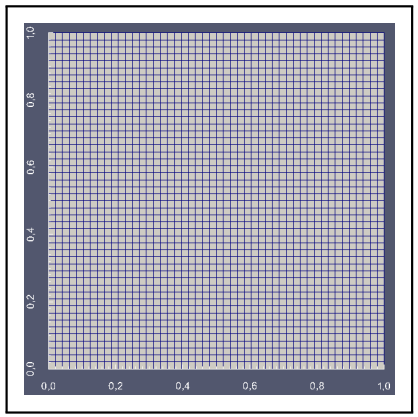
$$\sum_{i=1}^{N} x_i \underbrace{a(\psi_i, \psi_j)}_{=:a_{ij}} = \underbrace{b(\psi_j)}_{=:b_j} \qquad \forall j \in \{1, \ldots, N\}$$

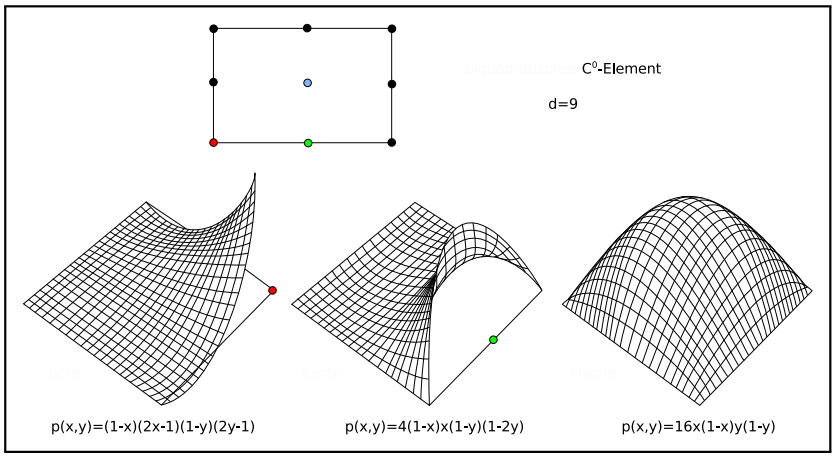
$$\boxed{\Rightarrow Ax = b}$$

Basic course content

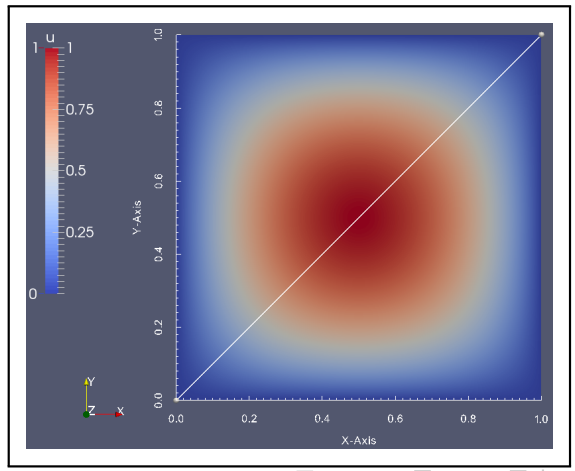


⇒

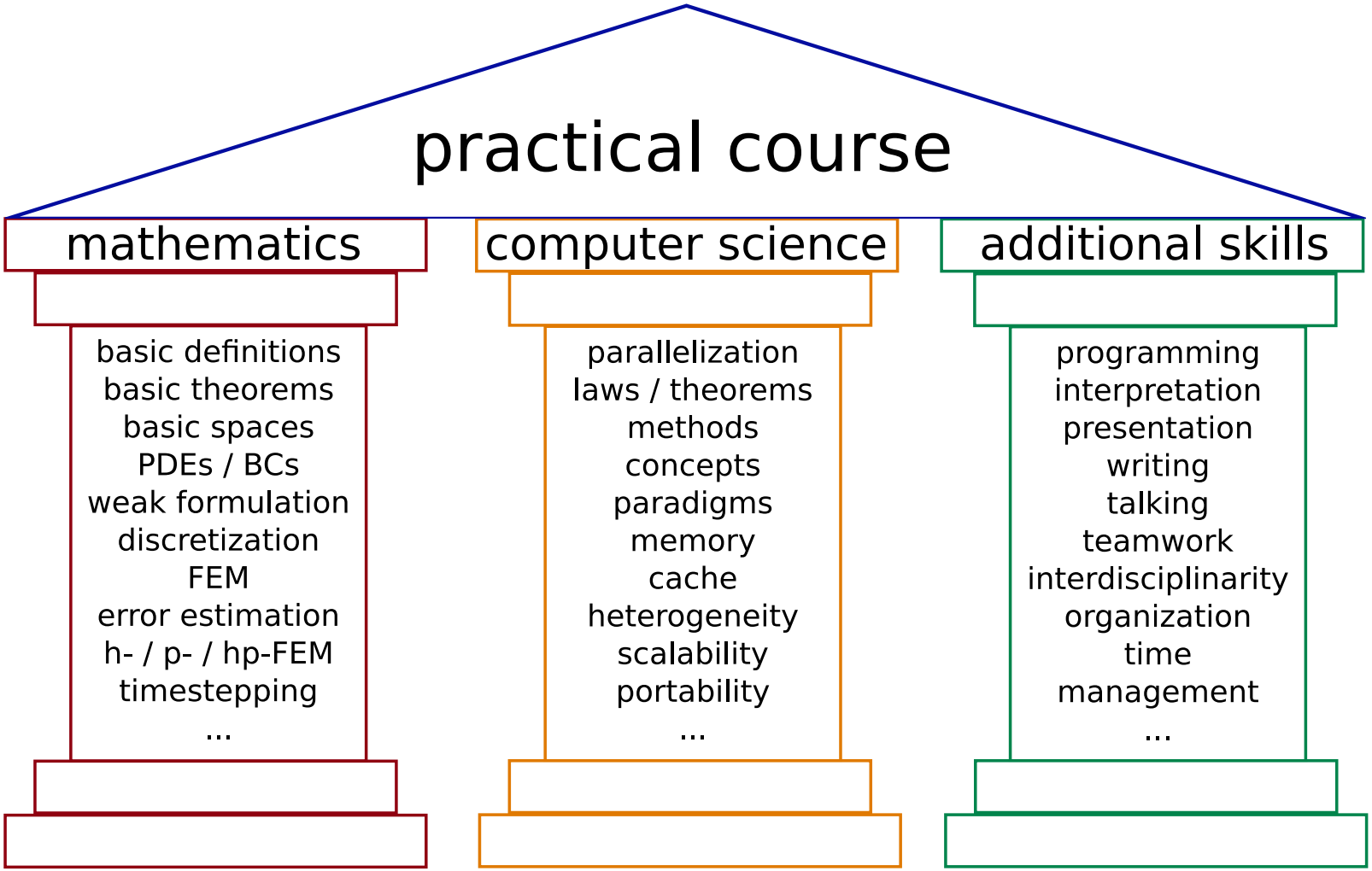




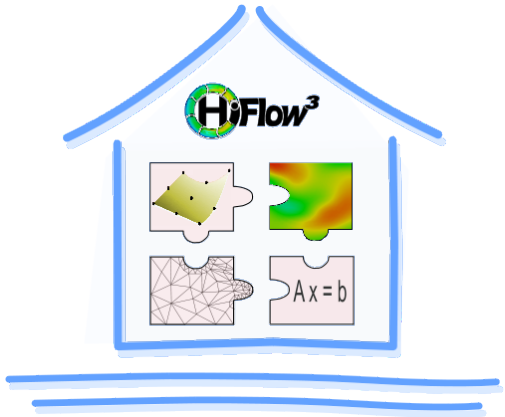
⇒



Basic course content



Used software

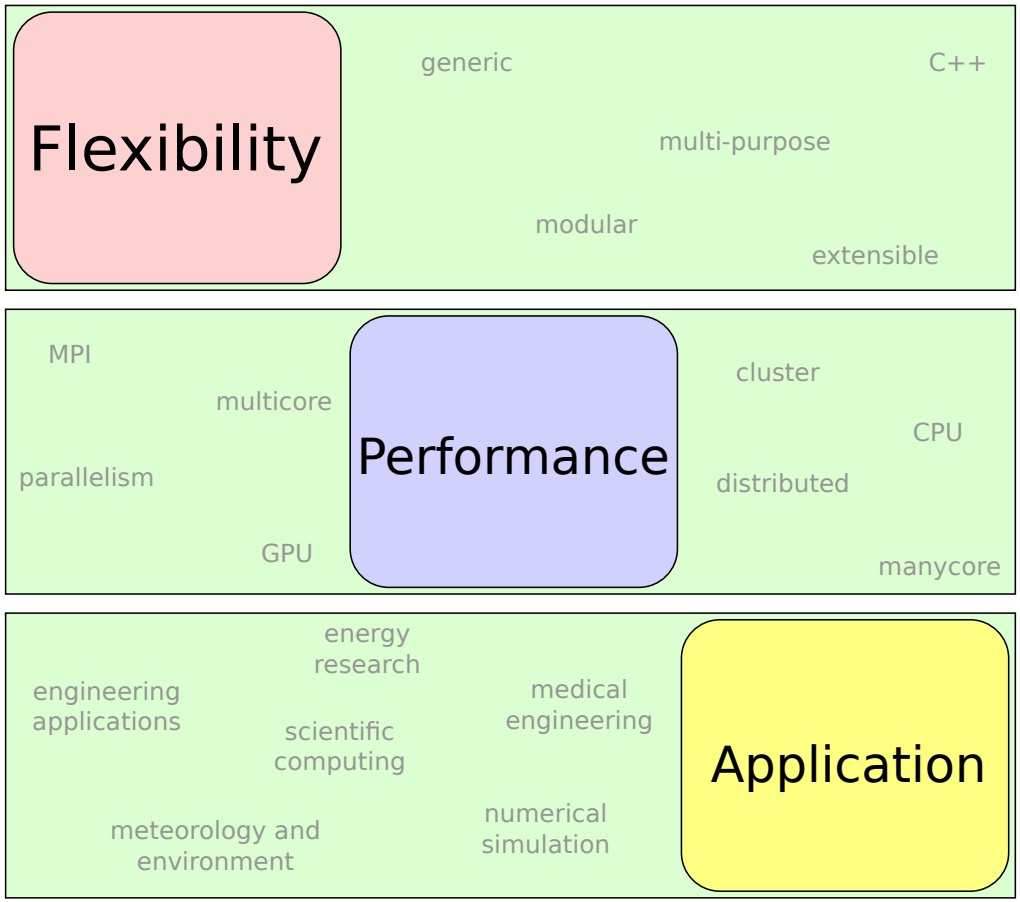


source: hiflow3.org

- 14 years of development and experience
- Open Source Software (LGPLv3-license)
- Programming language: C++
- For large scale problems modelled by PDEs
- Discretization in HiFlow^3 with Finite Elements
- Tools for solving efficiently and accurately

Used software

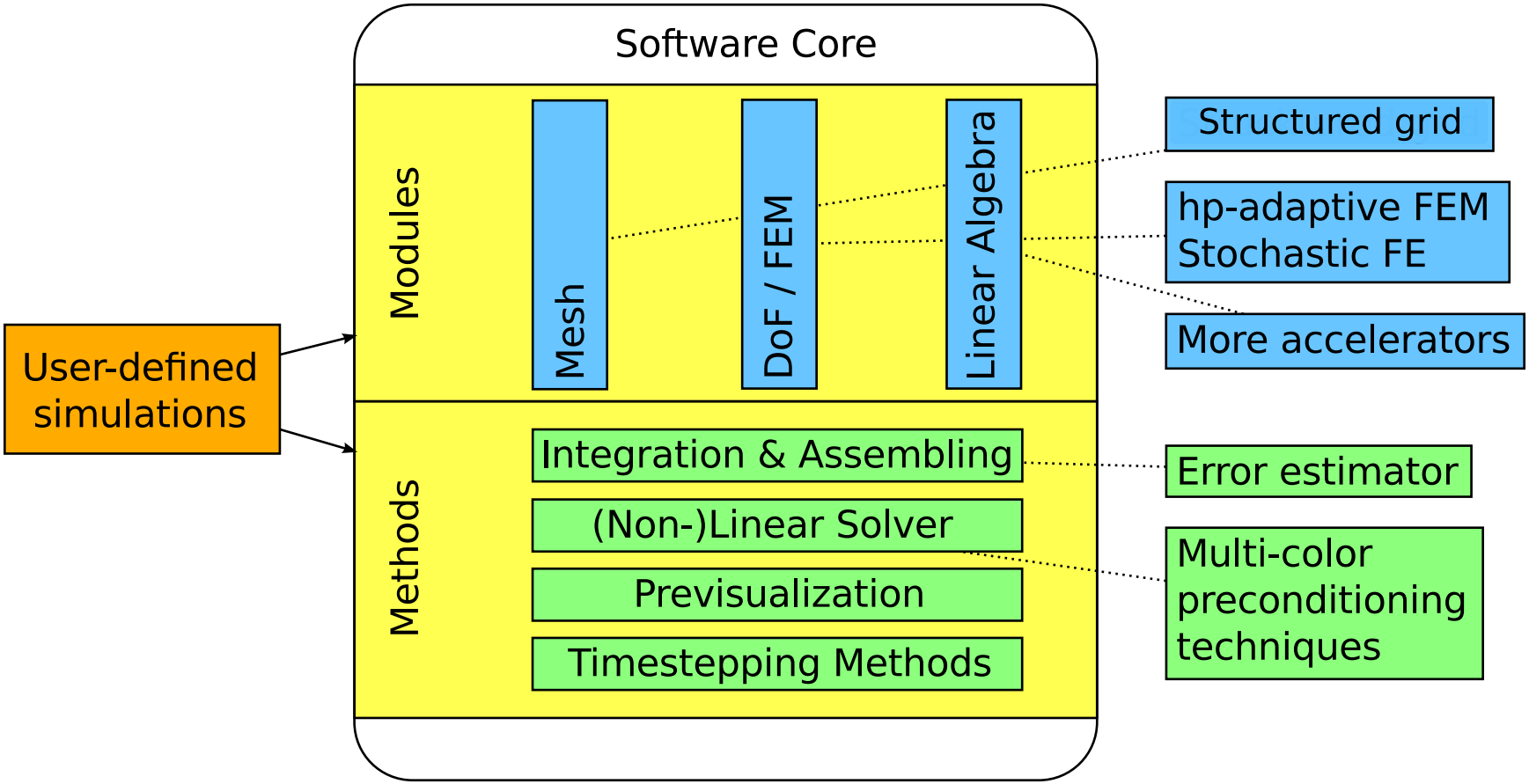
HiFlow^3: Concept



source: hiflow3.org

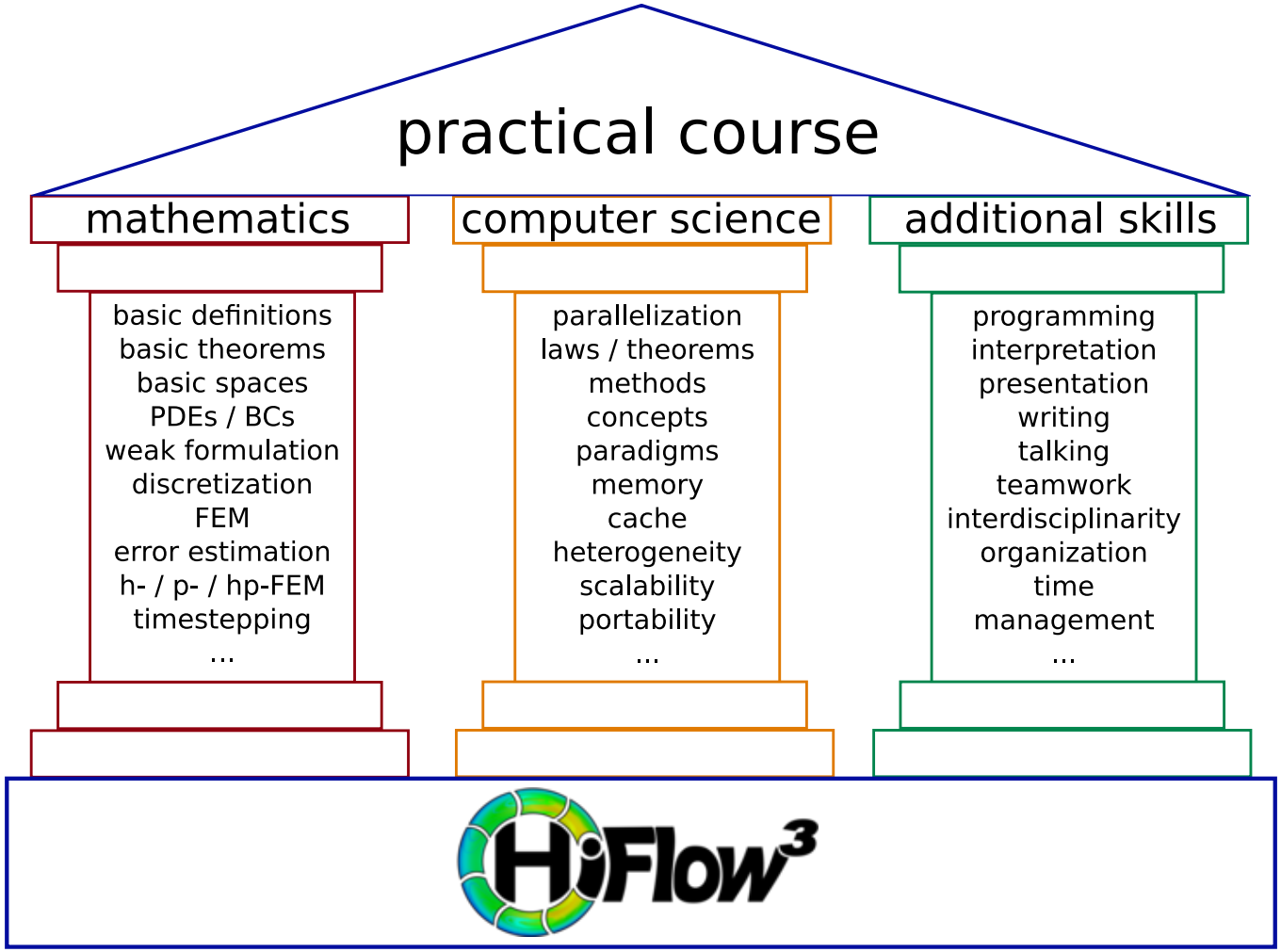
Used software

HiFlow^3: Structure

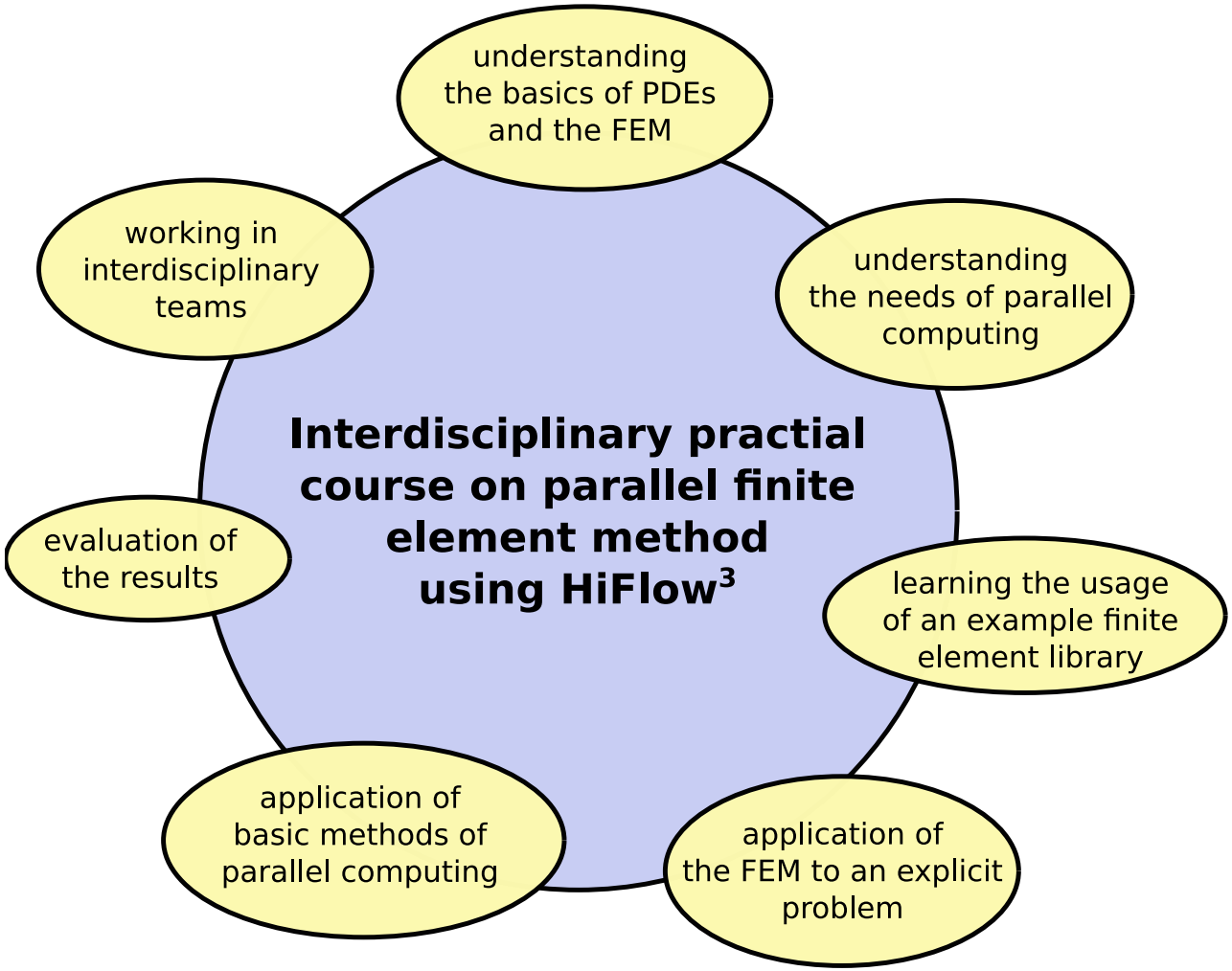


source: hiflow3.org

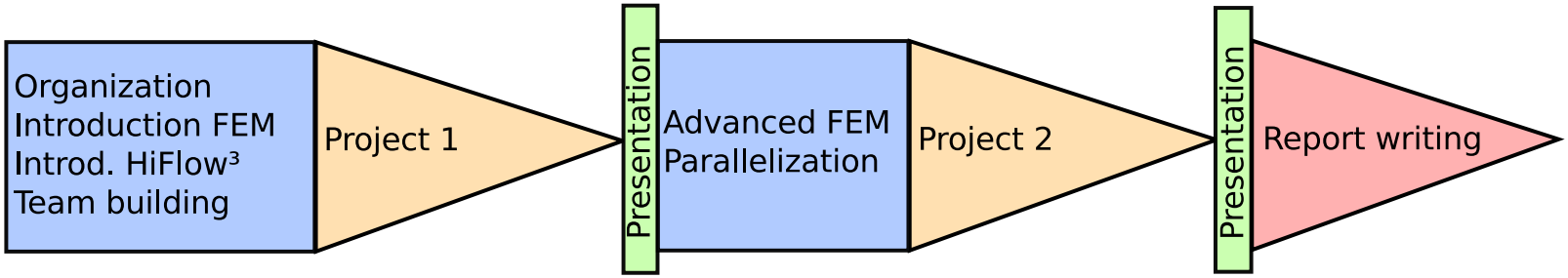
Basic course content



Learning objectives

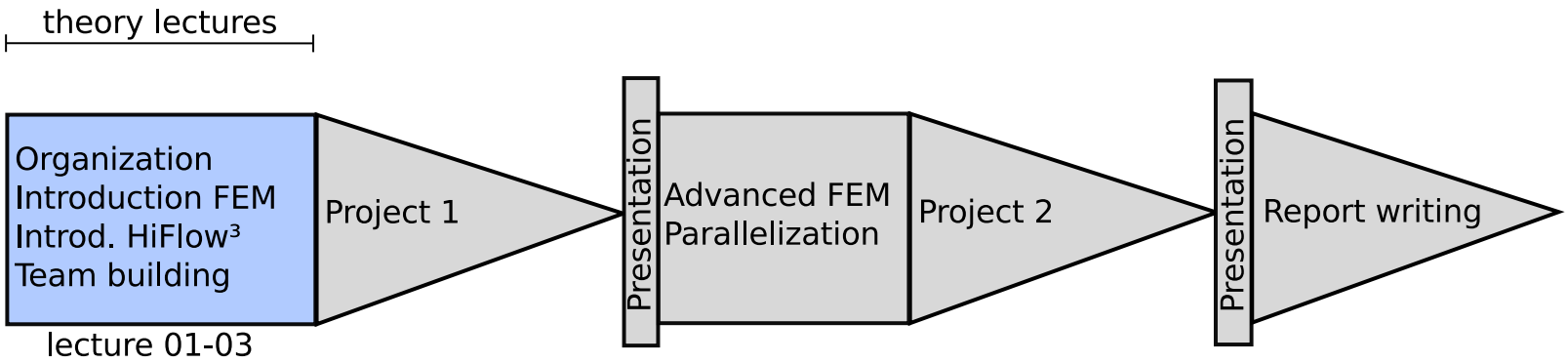


Practical application



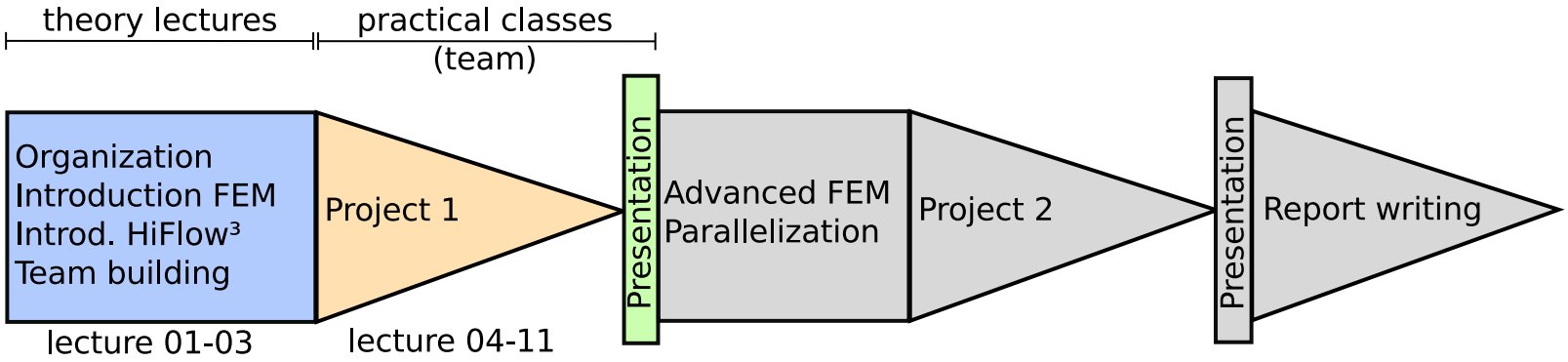
- Organized in theory lectures and practical classes
- 2 main sections + report writing
- Presentations and discussions
- Interdiscipinary: no prerequisites

Practical application



- Basic definitions / theorems / spaces
- PDEs / BCs / weak formulation
- Stationary h- / p- / hp-FEM
- Laws on parallelization
- Basic parallelization methods / concepts / paradigms
- HiFlow^3

Practical application

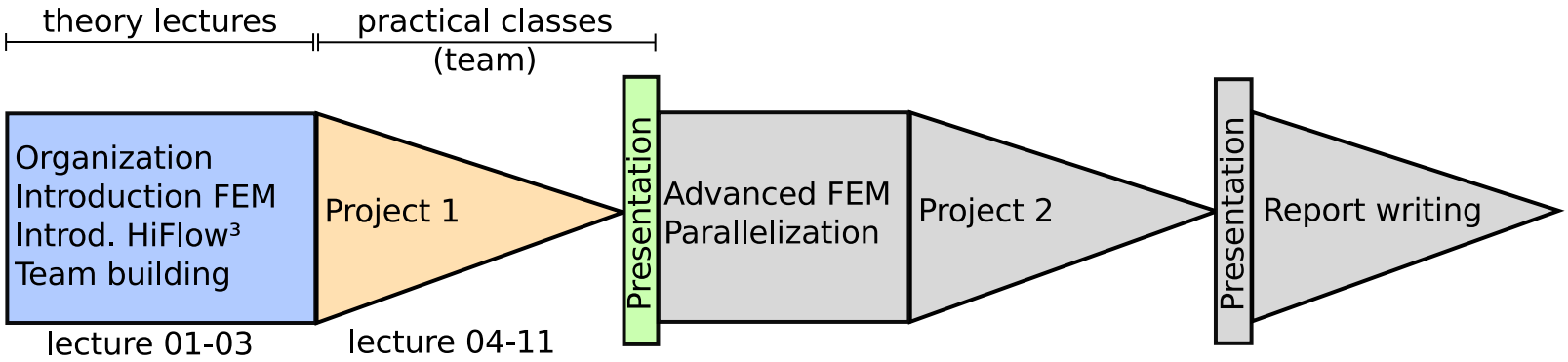


- Poisson's equation or similar
- Exercise sheets, no compulsory attendance

Exercises

1. Derive the variational formulation of the model problem. What assumptions must be made on u and f to be well-posed?

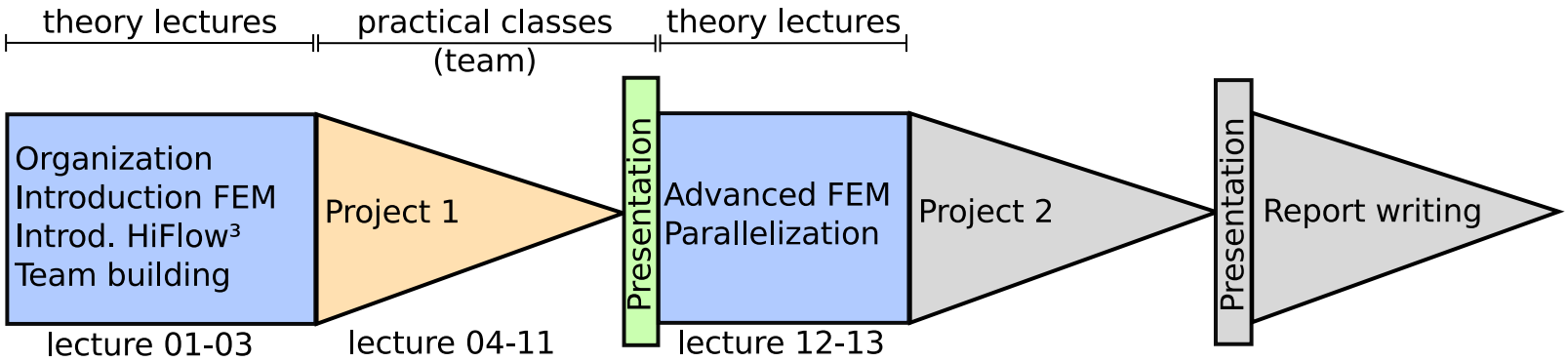
Practical application



Exercises

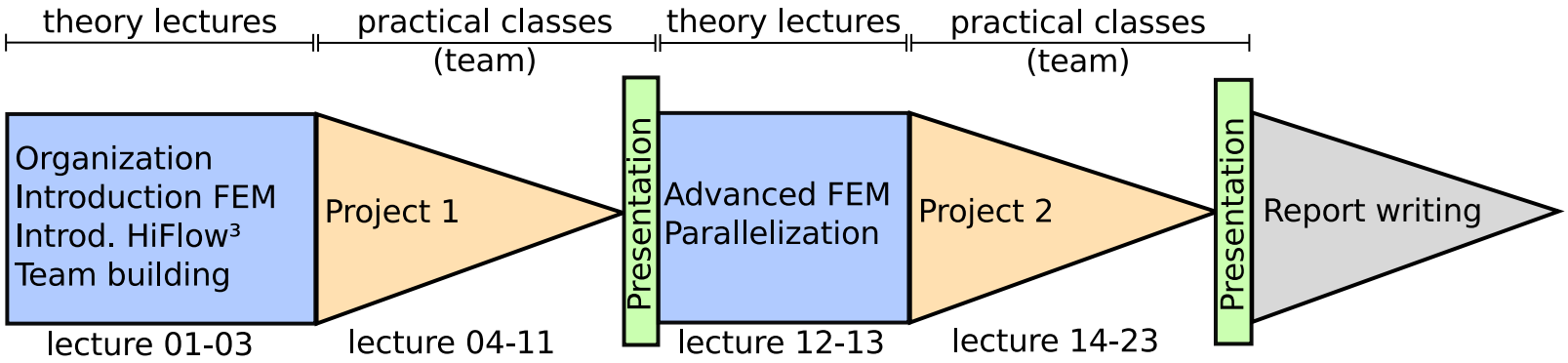
2. Complete the provided code skeleton to solve the variational problem with linear finite elements. Add a loop to perform uniform mesh refinement.
3. Define “speedup“ and “efficiency“. What is the difference of the paradigms of MPI and OpenMP?

Practical application



- Instationary FEM
- Special Finite Elements
- Advanced mathematics (preconditioning, stability problems, . . .)
- Memory, caches, scalability
- Data organization / domain decomposition methods
- Load balancing / task scheduling

Practical application

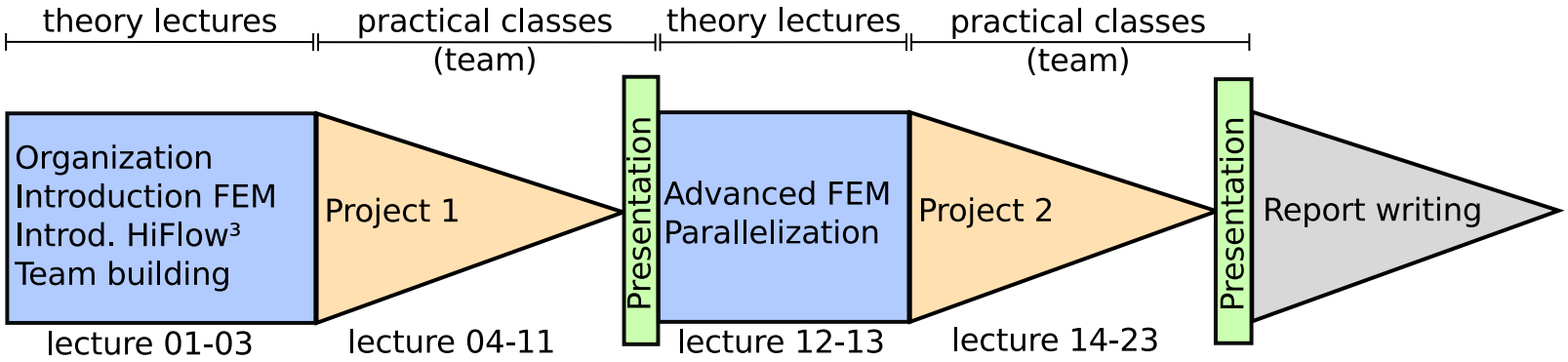


- Incompressible Navier-Stokes equations or similar
- Wishes of participants can be taken into account

Exercises

1. Investigate the value of Pe [Conv.-Diff. equ.] with respect to the stability of the solution! What is the maximum value of Pe yielding a stable system?

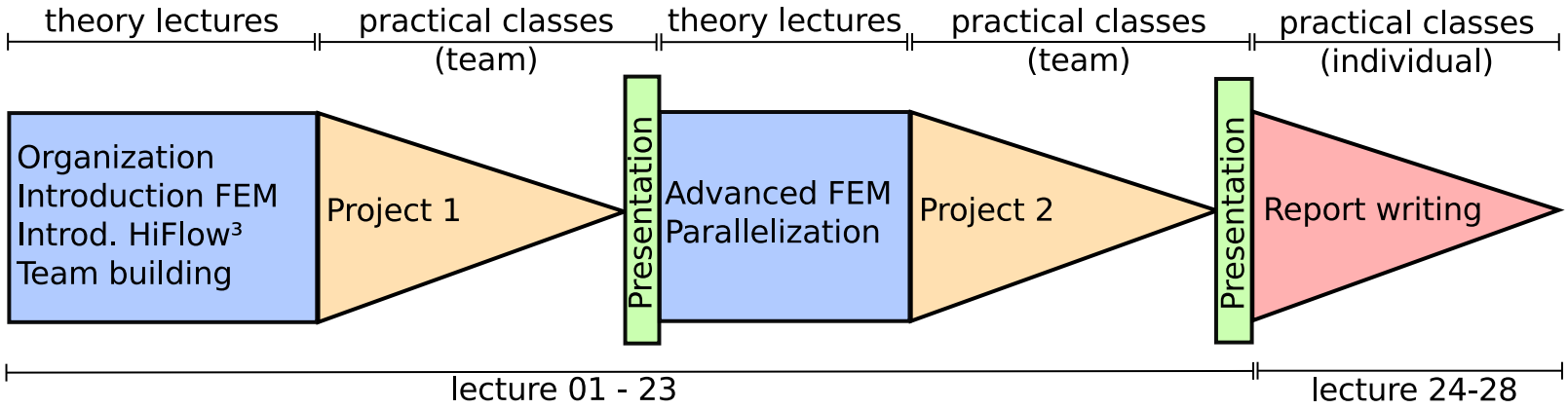
Practical application



Exercises

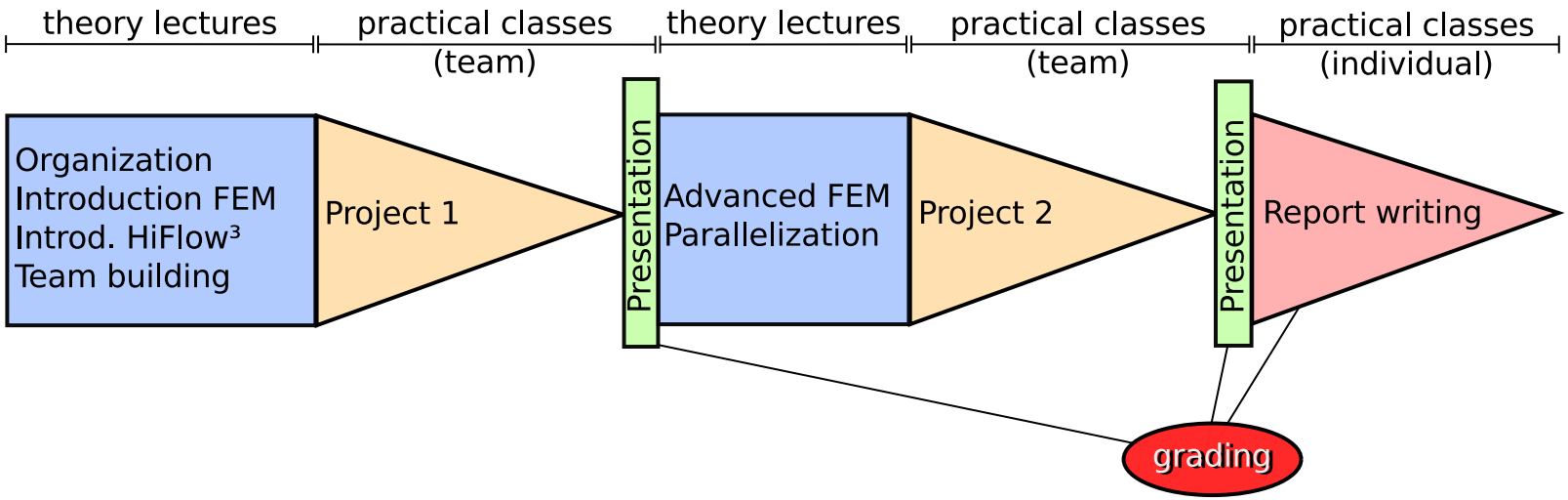
2. Implement either a fractional-step method (e.g. fractional-step-θ scheme) or a higher-order time-stepping scheme (e.g. fourth order Runge-Kutta method).
3. Find an efficient way to parallelize the SSOR preconditioning method. Name the used levels of parallelism.

Practical application



- Individual report
- Work and results of both projects
- 10-15 pages (without pictures, title page, references, . . .)
- Extensible time range (max. 4 weeks)

Practical application



UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Karlsruhe Institute of Technology

Statistics

- *Duration:* 14 weeks, 2 x 90-minute-lectures per week (1 period)
- *Working hours:* 120
- *Credits:* 4 ECTS
- *Participants:* up to 7 groups (3 students per group)
- *Personnel costs:* 2 teaching assistants
- *Environment:* HPC-Cluster / multi- or many-core architecture(s)

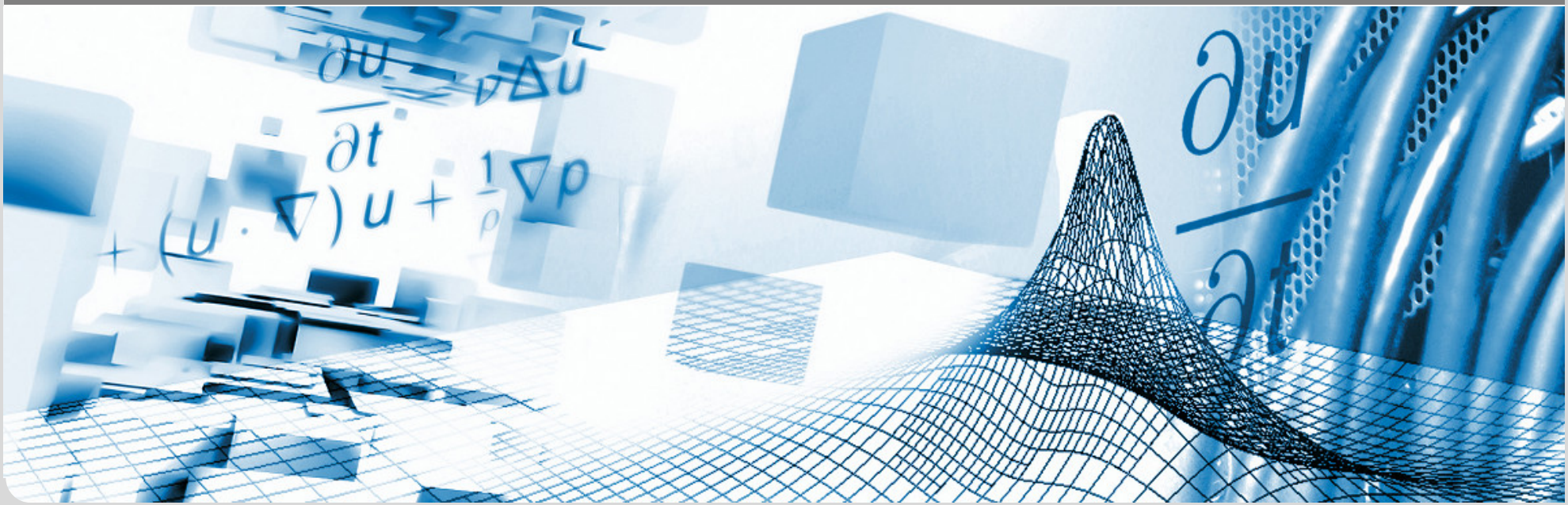
Karlsruhe Institute of Technology

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

Interdisciplinary practical course on parallel finite element method using HiFlow^3

E. Treiber, S. Gawlok, M. Hoffmann, V. Heuveline, W. Karl | EuroEDUPAR, 2015/08/24

KARLSRUHE INSTITUTE OF TECHNOLOGY - ITEC/CAPP, HEIDELBERG UNIVERSITY - IWR/EMCL



References I

http://www.avs.com/

Balay, S., Abhyankar, S., Adams, M. F., Brown, J., Brune, P., Buschelman, K., Eijkhout, V., Gropp, W. D., Kaushik, D., Knepley, M. G., McInnes, L. C., Rupp, K., Smith, B. F., Zhang, H.: PETSc Web page. http://www.mcs.anl.gov/petsc (2014)

Balay, S., Abhyankars., Adams, M. F., Brown, J., Brune, P., Buschelman, K., Eijkhout, V., Gropp, W. D., Kaushik, D., Knepley, M. G., McInnes, L. C., Rupp, K., Smith, B. F., Zhang, H.: PETSc Users Manual. Argonne National Laboratory, ANL-95/11 - Revision 3.5. http://www.mcs.anl.gov/petsc (2014)

Balay, S., Gropp, W. D., McInnes, L. C., Smith, B. F.: Efficient Management of Parallelism in Object Oriented Numerical Software Libraries. Modern Software Tools in Scientific Computing, E. Arge and A. M. Bruaset and H. P. Langtangen, 163–202, Birkhäuser Press (1997)

Heuveline, V., et. al.: HiFlow^3: A Hardware-Aware Parallel Finite Element Package. Tools for High Performance Computing 2011, Springer, 139â-151 (2012)

Heuveline, V., Ketelaer, E., Ronnas, S., Schmidtobreick, M., Wlotzka, M.: Scalability Study of HiFlow^3 based on a Fluid Flow Channel Benchmark. Preprint Series of the Engineering Mathematics and Computing Lab (EMCL) (2012)

http://www.hiflow3.org/

http://www.ilias.de/

Karypis, G., Kumar, V.: A Fast and Highly Quality Multilevel Scheme for Partitioning Irregular Graphs. SIAM Journal on Scientific Computing, Vol. 20, No. 1, 359â-392 (1999)

References II

Mayer, J.: ILU++: A new software package for solving sparse linear systems with iterative methods. PAMM, Proc. Appl. Math. Mech. 7, 2020123 – 2020124 (2007)

http://www.mpi-forum.org/

http://mumps-solver.org/

http://www.netlib.org/blas/

http://www.netlib.org/clapack/

Nickolls, J., Buck, I., Garland, M., Skadron, K.: Scalable Parallel Programming with CUDA. ACM Queue, vol. 6 no. 2, 40–53 (2008)

Saad, Y.: Iterative Methods for Sparse Linear Systems. 2nd edition. Society for Industrial and Applied Mathematics (2003)

Schroeder, W., et al.: The Visualization Toolkit, 3rd Edition. Kitware, Inc. (2003)

Stone, J. E., Gohara, D., Shi, G.: OpenCL: A Parallel Programming Standard for Heterogeneous Computing Systems. IEEE Design & Test, Volume 12, Issue 3, 66–73 (2010)

http://www.vtk.org/

http://www.xdmf.org/