**Scientific Computing, bridging course**

**Introduction**

Stefan Pålsson, stefan.palsson@it.uu.se

---

**Teacher**

Stefan Pålsson, stefan.palsson@it.uu.se
Office: ICT 2309
Phone: 070 425 0213
+ TA:s on the labs

---

**Bridging?**

Bridging the gap between what you know and what you need to know (in scientific computing)

Preparing for Scientific Computing III and higher courses

---

**Learning outcomes (goals)**

To pass, the student should be able to:

- describe key concepts covered in the course (see Content) and perform tasks that require knowledge about these concepts;
- in general terms explain the ideas behind, and be able to use algorithms for solving linear systems, ordinary differential equations and for Monte Carlo simulations;
- analyse properties of the computational algorithms and mathematical models using the analytical tools presented in the course;
- discuss suitable methods and algorithms given a application problem

Furthermore

- given a mathematical model, solve problems in science and engineering by structuring the problem, choose appropriate numerical method and generate solution using software and by writing programming code;
- present, explain, summarise, evaluate and discuss solution methods and results.

Examined through assignments and workouts

Grades: U/G (fail/pass)

---

**Text book**

Michael T. Heath
Scientific Computing: an introductory survey
2nd ed. McGraw-Hill
Some questions

www.menti.com
code: 98 45 29

Scientific Computing?

• How can we foresee the path of Irma?
• Why do the plume get wider and wider?

Scientific Computing?

• Simulation with ensemble methods (some kind of Monte Carlo method)

Scientific Computing?

In general

• Computational methods and algorithms, problems that can’t be solved by hand or the ‘mathematical’ way.
We typically solve HUGE problems, i.e. protein structure prediction.

• What method fit a particular problem? Keep control of the errors
• Software and programming
The course structure

- Four themes

<table>
<thead>
<tr>
<th>1a Introduction: Matlab, Programming in Matlab</th>
</tr>
</thead>
<tbody>
<tr>
<td>1b Introduction: Computer arithmetic and errors</td>
</tr>
<tr>
<td>2 Linear Equations</td>
</tr>
<tr>
<td>3 Ordinary Differential equations</td>
</tr>
<tr>
<td>4 Monte Carlo methods</td>
</tr>
</tbody>
</table>

Each block begin with a computer lab

- Computer lab
- Lectures
- Workout
- Problem solving

- All parts are tightly tied together
- Mandatory to do workout and assignments in time (other parts not mandatory)

But hey, let’s move on to the computer lab

- Log on to the Student Portal and find course page
- Find Lab Exercises in left margin
- Do Lab 1, Intro till Matlab (two parts)

Nothing to hand in! 😊