

Information technology

Studies in English



This course is a part of an undergraduate programme taught fully in English. For more on studying in English at Cracow University of Technology please visit <http://www.civ-eng.pk.edu.pl>

Moodle for Information Technology

Experimental module in Moodle can be found at <http://elf2.pk.edu.pl/course/view.php?id=2024>

Course description

The course will enable students to comprehend the concepts and practice of Computer Science, the main focus is however put on developing programming skills. The main environment to exercise programming concepts Matlab. The minor goal is to give the student wider perspective on operating systems and software tools for data processing, visualization, geometric modelling. The student should also develop appreciation of the implications of computer use in solving engineering and scientific problems.

Duration:	1 semester
Number of hours:	15 (lectures) + 15 (laboratories). The lectures and laboratories are every two weeks (90 min. each)
Instructors:	dr inż. Roman Putanowicz (lecturer)

Assessment method

The computer labs grade is the same time the course grade.

Computer lab grading rules

- Students prepare to computer lab in their own study time.
- It is allowed to miss only one computer lab.
- The grade for computer labs is based on student's engagement, activity, progress and the results of a test written during the last lab.
- Students who failed to get passing grade for computer labs have one additional resit term for the test. The data for the resit term will be depending on the circumstances (generally one date for all students).

Important

- Generally I distinguish just two groups of students: the one who want to learn something and the others. Everything else are just less important details.
- The accuracy and objectivity of grading depends in fact on how much feedback on your engagement and achieved results you provide.
- **All the time it is possible to discuss the marks.**

Grading table

Effects	Engagement		
	small	medium	big
small	2.0	3	3.5
medium	3.5	3.5	4
big	4	4.5	5

On-line resources

All course materials are available on-line from the course web page:

- <http://www.l5.pk.edu.pl/~putanowr/iten.html>

The materials on the above page are available also in PDF format: please search for the “Export PDF” link in the toolbox in the left panel.

Lectures

Lecture	Title	Content
1	Introduction to the course. PDF	Aim and scope. Method of working. Motivation. Requirements. Overview of software tools. Sample problems.
2	Introduction to programming. PDF	Processor and program. High level programming languages. Compilation versus interpretation. Overview of popular languages. Introduction to programming in Matlab/Octave.
3	Diving into Matlab/Octave programming	Pseudocode. Variables and assignment operator. Matrix variables. Conditional statement. Loops. Functions. Programming paradigms: structural object oriented, functional
4	Data structures and algorithms	Basic data structures: arrays, lists, sets, graphs. Overview of selected algorithms for sequences. Sorting and searching. Matrix algorithms.
5	Data formats and data processing	Binary code. Data types. Characters, ASCII, code pages, UTF-8. Data formats. HDF5, JSON, XML, CSV. Data visualization. ParaView, Matplotlib, plotting in Matlab and Octave.
6	Introduction to geometric modeling	Geometric models and their representations. B-Rep i CSG. Meshes as data structures. Basic geometric transformations. Selected tools: GMSH, OpenScad, Blender.
7	Overview of software engineering	Repositories. Git, GitHub. Documenting code. Markdown. Linux environment. Automating repetitive tasks. Software licences.

Lecture	Title	Content
8	Summary	The role of programming in engineering curriculum.

Computer labs

No	Title
1	Introduction to GNU/Linux operating system
2	Getting started with Octave
3	Defining and using functions
4	Control flow : loops and conditional statements
5	More on control flow; Octave versus Matlab
6	Solving problems; more on plotting
7	Solving problems; operating on files
8	Summary



Self-checking projects (not mandatory)

Some students may wish to check their programming skills by solving problems more complex than the ones presented during the labs. Some ideas for such problems are presented below. Students are encouraged to extend the projects with their own ideas. **These projects are not mandatory.** Students wishing to have their solution to be checked, should prepare a report as indicated at each project sheet.

The report should be prepared as PDF file and send by e-mail to the respective tutor. There are no specific requirements on the document preparation system. One can use

- LaTeX - [please see the notes](#)
- Microsoft Office,
- web based solutions (raw HTML or some wikis and then printing),
- other

No	Project description	Assignment sheet
1	Trajectory of the mass center of three points	mass_centre_project.pdf
2	Velocity of a point moving along given trajectory	trajectory_project.pdf
3	The first and the second derivative of a function	derivatives_project.pdf
4	The shape of a deforming circle	deforming_circle_project.pdf
5	The sum of a function series	adding_signals_project.pdf
6	The volume and area surface of a solid	volume_surface_project.pdf



Reading list

Primary readings

- Primer, MathWorks, 2019 [PDF](#)

- David Houcque, Introduction to Matlab for Engineering Students, Northwestern University, 2005, [PDF](#)
- Tony Gaddis, Starting out with Python, Pearson, 2017 (4th edition)

Additional readings

- W. Gander, Learning MATLAB. A problem Solving Approach, Springer, 2015
- K.D Lee, Python Programming Fundamentals, Springer, 2011
- J. D. Foley et al, Computer Graphics: Principles and Practice, Addison Wesley; 3rd Edition, 2013
- D. M. Bourg, Physics for Game Developers, O'Reilly, 2001



From:

<https://cce.pk.edu.pl/~putanowr/dokuwiki/> - Roman Putanowicz Wiki

Permanent link:

<https://cce.pk.edu.pl/~putanowr/dokuwiki/doku.php?id=en:teaching:subjects:it&rev=1585310698> 

Last update: **2020/03/27 13:04**