Assignment 0 – CV1 WS16/17 17/10/2016

Prof. Stefan Roth
Junhwa Hur
Tobias Plötz

This assignment is due on October 31th, 2016 at 12:00.

Group work and grading policy

You are required to work on each assignment in groups of ***two*** people. It is up to you to form groups, but please note that the group assignments cannot change after the first homework assignment. If you can not find a group partner on your own, we will assign you to another person.

Programming exercises

For the programming exercises you will be asked to hand in Julia code. Please use version **v0.5.0** of Julia as we will use it to grade your solution. You need to comment your code in sufficient detail so that it will be easily clear to us what each part of your code does.

Your code should display your results so that we can judge if your code works from the results alone. Of course, we will still look at the code. You **must** adhere to the naming scheme for functions and files included with each problem.

Do *not* rename function files and do *not* change the given function signatures. If you feel that there is a mistake in the assignments, ask us on Moodle.

Pen & paper exercises

You might also have some theoretical exercises to do in the assignments. In this case we would greatly appreciate if you could typeset the theoretical part of your solution (ideally with LaTeX) and submit it as a PDF along with the rest of your solution. If you are not sufficiently familiar with a mathematical typesetting system such as LaTeX, you can also hand in a high resolution scan of a handwritten solution. Please write neatly and legibly.

Files you need

All the data you will need for problems will be available on Moodle `https://moodle.tu-darmstadt.de/course/view.php?id=8060`.

Handing in

Please upload your solutions to Moodle. You only need to submit one solution per group. If, for whatever reason, you cannot access Moodle get in touch with us as soon as possible.

Upload all your solution files (the writeup, `.jl` files, and other required files) as a single `.zip` or `.tar.gz` file. **Please note that we cannot accept file formats other than the ones specified!**

Late handins

We will accept late handins, but we will take *20% of the total reachable points* off for every day that you are late. Note that even 1 minute late will be counted as being one day late! After the exercise has been discussed in class, you can no longer hand in.

Other remarks

Your grade will depend on various factors. Of course it will be determined by the correctness of your answer. But it will also depend on a clear presentation of your results and good writing style. It is your task to find a way to *explain clearly how* you solved the problems. Note that you will get grades for the solution, not for the result. If you get stuck, try to explain why and describe the problems you encountered – you can get partial credit even if you did not complete the task. So please hand in enough information for us to understand what you did, what things you tried, and how it worked! We will provide skeleton code for every assignment. Please use the provided interfaces as it allows us to better understand what you did.

We encourage interaction about class-related topics both within and outside of class. However, you should not share solutions with other groups, and **everything you hand in must be your own work**. You are also not allowed to plainly copy material from the web. You are required to **acknowledge any source of information that you used to solve the homework** (i.e. books other than the course books, papers, etc.). Acknowledgements will *not* affect your grade. Thus, there is no reason why you would not acknowledge sources properly. Not acknowledging a source that you have used, on the other hand, is a clear violation of academic ethics. Note that the university as well as the department is very serious about plagiarism. For more details please see `http://www.informatik.tu-darmstadt.de/index.php?id=202` and `http://plagiarism.org`.

Problem 1 - Getting to know Julia (1 points)

In this task you will set up Julia on your system and learn how to install additional packages. Two packages that you will often use are **JLD** for saving and loading data as well as **PyPlot** for plotting your results.

Tasks:

- Download and install Julia **v0.5.0** from `http://julialang.org/downloads/`.
  You can use the Juno IDE (`https://github.com/JunoLab/uber-juno/blob/master/setup.md`) for developing and testing your code.
- Install the packages **JLD** and **PyPlot**. Instructions on how to install a package for Julia can be found at `http://docs.julialang.org/en/release-0.5/manual/packages/`.

  Note: PyPlot requires a Python installation.

To familiarize yourselves with Julia, please implement the following toy problem:

- Function *loadImage* to load the given image.
- Function *saveFile* to save an array to a .jld. Use the key "img" for saving.
- Function *loadFile* to load the previously saved file again.
- Function *mirrorHorizontal* to mirror the image along its y-axis.
- Function *showImages* to show both images in one plot using PyPlot.