Assignment 0

To be assigned to an exercise group,
you must complete Task 4 by Wednesday, April 26, 2017, 21:00.
Submit all other results by Friday, April 28, 2017, 23:59.

For help, contact alp-staff@lists.iai.uni-bonn.de.

You must perform each task on this assignment sheet <u>yourself</u>.
Group work will start with Assignment 1.

Task 1. *Install SmartGit* (2 Points)

Download and install SmartGit from http://www.syntevo.com/. Go slowly through the installation steps so that you really read what you are asked!!! Make sure that you choose "non-commercial license" when you are asked for the type of license that you want!!!!

▶ You get the points if you demonstrate to your tutor next week that you have a working installation (either on your laptop or on a computer in the pool) and show them the line you added to the poem (see task 3).

Task 2. *Create a public key for our Git server* (4 Points)

Please follow the steps explained in

https://sewiki.iai.uni-bonn.de/teaching/lectures/alp/2017/infrastructure_git

to create a public ssh key. Then send the generated public key file as an e-mail attachment to alp-staff@lists.iai.uni-bonn.de. Do not paste the key into the e-mail!!!

▶ **You get the points if your contribution is pushed to the repository before the deadline mentioned in red at the top. This requirement applies to all the following tasks, so it will not be repeated again**.

Task 3. *Hello via Git* (6 Points)

To test your Git installation, start by downloading the private ssh key for the dummy user "alp.anonymous" from the Section "Git Repositories for Your Assignments" of

https://sewiki.iai.uni-bonn.de/teaching/lectures/alp/2017/infrastructure_git

Use it to clone

ssh://git-se@git.iai.uni-bonn.de/alp2017_shared

Then open the "poem.txt" file that is contained in the project and
- add a line to the poem (or more if you feel creative ;-)),
- commit (don't forget to add a comment, otherwise Git won't let you commit),
- pull
- resolve conflicts, if any
- and finally push your change.

If the file on the Git server was changed during your editing, Git will merge the changes when you pull. If somebody edited exactly the same line, Git will indicate a conflict after pulling and you will need to resolve the conflict manually (e.g. by using the side-by-side compare view that opens when double-clicking the conflicting file in SmartGit) then commit and push.

Task 4. *Availability data for your group assignment* (10 Points)

We need your availability data as Prolog facts. In the " alp_shared" repository (see previous task), you find a folder called "timetable" containing the files "participants.pl" and "student_availability.pl".

Please add to "participants.pl" a fact that tells us your e-mail address and your preferred team-mates and to "student_availability.pl" a set of facts that tell us when you are available. The format of the facts is explained in the files.

▶ **Your assignment to an exercise group depends on the data that you submit!**
▶ If you do not submit in time we will not assign you to any group.
▶ If you submit in time but we cannot fulfill your preferences you can change your group later (see website for instructions how to do it).

Task 5. *SWI-Prolog* (4 Points)

Install SWI-Prolog on your computer. You can download it at

> http://www.swi-prolog.org/

Ensure that SWI-Prolog is contained in the "Path" variable of your OS environment.
Then
- Create a prolog (.pl) file containing the code of the "ancestor" example from the lecture and complement it with facts for your own family.
- Use the SWI-Prolog console to run a query that asks for your own ancestors.
- Make a screenshot showing the "ancestor" query and its results in the SWI-Prolog console.

▶ To get the full points Name the screenshot "myFirstName_mySecondName_task6", store in in the folder "assignment 0" and push it to the git repository before the deadline.

▶ If you don't submit in time via Git you can still get half the points by showing the query running on your computer (or on your Windows account) during the tutorial.

Task 8. *Install the PDT* and JTransformer (4 Points)

Perform the installation(s) described at

https://sewiki.iai.uni-bonn.de/teaching/lectures/alp/2017/infrastructure_prolog

Then do the same as in Task 6 but this time using the PDT.

▶ Create and push a screenshot showing the ancestor example running in the PDT console.