

# STL, Homework 1

## due 18.03.2024, 08:00

### Prerequisites

- Basics
  - Your programs must have been written and tested on Janus
  - Use subdirectories \$HOME/Exercises/HW1/1, \$HOME/Exercises/HW1/2, \$HOME/Exercises/HW2/3
  - Write your codes in C (preferred, other languages upon request), use GCC
  - No global variables
- Building
  - Include the following three arguments to compile (here: C):  
-Wall -std=iso9899:2018 -pedantic
  - No warnings should be issued by the compiler
  - Use Makefiles
    - One for each exercise, no arguments (simply “make” should be sufficient for building)
    - Compilation and linking must be separate
    - Only files that have been altered shall be rebuilt.
- Submitting
  - Do not include object files, executables, libraries
  - Create a single xz'ed tar file of the following form:  
<last name>\_<matriculation number>.tar.xz
  - Upload only this single archive file to Moodle
- Presentation
  - You must be ready to present and discuss your submission
  - The files on Janus must match with your submission on Moodle

### Exercises (10 points in total)

- **1:** (2 points) Write a program in C (preferred) which includes a function “vec\_mul” that multiplies each entry of a given vector of numbers by a given factor; use a pointer of data type “double \*” to address the vector. Demonstrate its correct functionality (hardcoded values are fine).  
Include the following files in your submission:
  - Makefile
  - Source code
- **2:** (3 points) Use Gnuplot or another “scriptable” software to generate a figure which looks similar to the one on page 14, Figure 5 (b) of (Khan et al, 2021: <https://doi.org/10.1145/3432261.3432263>); create the data file(s) yourself; the output file must be in pdf, a Makefile must generate the pdf.  
Include the following files in your submission:
  - Makefile to generate this figure in pdf
  - (Gnuplot) script file, (Gnuplot) data files
  - Figure in pdf
- **3:** (5 points) Demonstrate the functionality (e.g., wall clock time, CPU time, cache misses, ..) of *The Performance Application Programming Interface* (PAPI, <https://icl.utk.edu/papi/>) on Janus; version 6.0.0 is installed on Janus - either use this or a newer version; hint: try to find an example code in the package.  
Include the following files in your submission:
  - Pdf that contains detailed steps including one or more screenshots
  - Make notes if some steps initially failed (e.g. broken library?)
  - Example source code