# Guide of CS402 Assignment 2

This assignment will be based on a different program, karman, which solves the Navier-Stokes equations on a structured grid.

## Getting Started

Download and extract a copy of karman from the CS402 web page. Inside the downloaded folder there is a Makefile and two subfolders: src and test. The src folder contains all the application source code, and the test folder contains some test input.

To compile karman type:

```
make
```

at the command line, and the executable will be placed in the top level directory.

### Output of Karman

Run the program in the normal way, and karman will generate a default problem and solve it, writing the final solution out to the file karman.bin.



### Karman options

The complete set of options that karman supports can be printed out using the following command.



```
./karman --help
karman. A simple computational fluid dynamics tutorial.
Usage: ./karman [OPTIONS]...
-h, --help      Print a summary of the options
-V, --version   Print the version number
-v, --verbose=LEVEL  Set the verbosity level. 0 is silent
-x,     --imax=IMAX     Set the number of interior cells in the X
direction
-y,     --jmax=JMAX     Set the number of interior cells in the Y
direction
-t, --t-end=TEND        Set the simulation end time
-d, --del-t=DELT        Set the simulation timestep size
-i, --infile=FILE       Read the initial simulation state from this
file
(default is 'karman.bin')
-o, --outfile=FILE      Write the final simulation state to this file
(default is 'karman.bin')
```



### View the karman solution

You can take a look at the solution by using the bin2pmm program built as part of the karman build process. The program works using standard input and output streams, so you need to run it like this:
`./bin2ppm < karman.bin > karman.ppm`

The ppm file should be viewable in any standard Unix image viewer, but you can use the program ppm2jpeg to convert it to a JPEG file if you prefer.

### Check the correctness of your parallel program

Once you start adding some code to karman, it's important that you check your results. You can do this with the diffbin program, that

compares the final solution from the vanilla karman to your new parallel solution:

```
./diffbin karman.vanilla.bin karman.bin
```

Note: Remember to rename the karman.bin file when you are doing multiple runs, otherwise it will be overwritten.

**Change the problem size**
As seen in the OpenMP lab, sometimes a larger problem size will give you better performance improvements. To create a bigger problem in karman, use the -x and -y flags to increase the number of cells in the X and Y dimensions.

## Profiling the code

To profile the code you will need to modify the makefile so that CFLAGS has a -pg at the end of it. This will instruct the profiler gprof to record timing behaviour for the main routines in the code.

Once you have changed the makefile then you need to rebuild the code using:

```
make clean all
```

When you run the program it will generate profiling information to gmon.out. This can be viewed using the command:



```
gprof  ./karman
```

this generates a sorted list of the execution time spent in each function.



another way to profile the code:



```
gcc -pg filename.c -c
gcc filename.o -pg -o filename
./filename
gprof filename > filename.gprof
```

This should give you a good idea as to where the majority of the runtime is spent. This is where you should focus your attention when thinking about parallelizing the code. We suggest you start your parallelisation of the program from the main computational part.

## The Main Loop

The timestep loop proceeds as follows:

- Calculate an appropriate time-step size.
- For each cell, compute a tentative new velocity based on the previous velocity and the stencil velocity values.
- For each cell, calculate the RHS of the pressure equation.
- For the entire pressure matrix, use Red/Black SOR to solve the Poisson equation. This takes a large number of iterations of the Red/Black process.
- For each cell, update the real velocity values based on the pressure matrix and the tentative velocity values.
- Update the boundary velocities.