Laboration 1
SF2568 Program construction in C++ for Scientific Computing

June 17, 2015

The aim of this lab session is to become comfortable with the C++ computing environment available at the Ubuntu machines. You can use an editor of your choice. I recommend to use emacs (and its C++ module).

Task 1 Implement, compile and run the Newton algorithm as presented in the lecture!

Task 2 The following task is taken from a course at Uppsala University.[1] Consider the program trace.cpp provided on the courses homepage. It consists of four functions:

- initialization (Get the dimension and return the square matrix)
- fill_vector (get a row of the matrix and fill it with random numbers from +10 to -10 and return the vector back to the main function)
- print_matrix (Get the dimension and display the matrix)
- diagonal_sum (Get the matrix, sum the diagonal values and return the sum)

Your task is to debug the program. Hint: Use the switch `-Wall` for `g++`!

[1] Courtesy Marcus Holm