Exercise 7 FYSY160 C++ numeerinen ohjelmointi Autumn 2011

return solution by 31.12.2011
Email the *commented* solution code (*.cpp) to :
address: `FYSY160(at)gmail.com` Subject-line: `Project work`
If you run into trouble, please send questions also to that address

1. Arc length of a function and Eigensystems

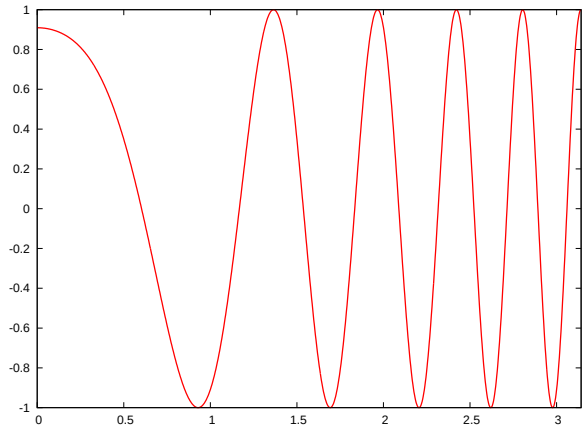
   The Arc-length of a function $f(x)$ between the interval $[a, b]$ can be calculated from the following equation

   $$s = \int_a^b \sqrt{1 + [f'(x)]^2} dx$$

   where $f'(x)$ is the derivative function of $f(x)$. Implement a C++ program that numerically calculates the arc length of a function

   $$f(x) = \sin(\alpha x^2 + \beta).$$

   between the interval $[0, \pi]$. Choose the parameters $\alpha = \pi$ and $\beta = 2$.



Kuva 1: Function $f(x) = \sin(\pi x^2 + 2)$ between $[0, \pi]$

You should get something like $s =$ `20.5026`. Calculate the derivative $f'(x)$ numerically (hint: exercise 4) and use GSL integration routine

(http://www.gnu.org/software/gsl/manual/html_node/QAG-adaptive-integration.html)

to evaluate the integral. Furthermore, pass both α and β as a data structure to the integral routine and pay attention to the modularity (= the method to split the program code into functions and subprograms) of the code, i.e. the derivative $f'(x)$ can be calculated in a separate function.

As a second separate task, implement a C++ program that finds the eigenvectors and eigenvalues of a Hermitian band matrix H of size 10×10.

$$\mathbf{H} = \begin{bmatrix} 0 & t & 0 & 0 & 0 & \cdots & 0 \\ t & 0 & t & 0 & 0 & \cdots & 0 \\ 0 & t & 0 & t & 0 & \cdots & 0 \\ 0 & 0 & t & 0 & t & \cdots & 0 \\ 0 & 0 & 0 & t & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 \end{bmatrix}$$

The Eigenvectors (x) and eigenvalues (λ) are the solutions to the equation

$$\mathbf{H}x = \lambda x$$

This type of matrix is very often used in condensed matter physics to model lattice systems and linear atomic chains. Set the t-parameter to be $t = -1.0$. Use Armadillo C++ library
(http://arma.sourceforge.net/docs.html) definitions for matrices and vectors and also to calculate the eigenvalues and eigenvectors. Print the results also to the screen. Check also that the eigenvectors are ortogonal by calculating the dot product of the first eigenvector with all the other eigenvectors. The dot products should give values very close to zero.
Study carefully the Armadillo documentation, you can find all the necessary commands and methods from there to build a working program easily.

Good luck!