1 NB

- ODE $\doteq$ Ordinary Differential Equation.
- Where appropriate reduce the argument (using periodicity, symmetry, reflection, complementary, reciprocal and other properites of the functions under consideration) to avoid numerical integrations over large intervals. Here is an example,

  ```
  double mysqrt(double x){
     assert(x>=0);
     if(x==0) return 0;
     if(x<1) return 1/mysqrt(1/x);
     if(x>16) return 4*mysqrt(x/16);
     if(x>4) return 2*mysqrt(x/4);
     /* calculate the result for x \in [1,4] */
     return result;
  }
  ```

  Other examples can be found in some of the formulations of the problems.
- Prove that you obtained the correct result (e.g. by comparing with function from `<math.h>`).
- You have to plot your functions in a range which is broader than the reduction range in order to prove that the reduction has been done properly.
- Check the supplied information with Wikipedia as I might have made a misprint (and/or find there — that is, in Wikipedia — the missing information should you need some).

2 Problems

0. **Natural logarithm (integral representation).** Implement a function that calculates the natural logarithm of a real positive number x using the integral representation

$$\ln(x) = \int_1^x \frac{1}{t} dt \,. \tag{1}$$

   Employ an integration routine from GSL. Before calling the integration routine reduce the logarithm of an arbitrary positive number to the logarithm of a number in the range $1 \leq x < 2$ using the formulae

$$\ln(x) = -\ln\left(\frac{1}{x}\right) \,,\ \ln(x) = \ln(2) + \ln(x/2) \,. \tag{2}$$

   Compare with the corresponding function from `<math.h>` or from GSL.

1. **Natural logarithm (ODE representation).** Do the same as in "Natural logarithm (integral representation)" but use the ODE representation,

$$\frac{dy}{dx} = \frac{1}{x} \,,\ y(1) = 0 \,, \tag{3}$$

   and one of the ODE routines from GSL.

   Compare with the corresponding function from `<math.h>` or from GSL.

2. **Natural logarithm (root finding).** Implement a function that calculates the natrual logarithm t of a real positive number x, $t = \ln(x)$, by solving the equation

$$e^t = x \,. \tag{4}$$

   You can use the function `exp` from `<math.h>` or `gsl_sf_exp` from GSL.

3. **Square root (ODE representation I).** Implement a function that calculates the square root of a real positive number x using the ODE representation

$$\frac{dy}{dx} = \frac{1}{2y} \, , \; y(1) = 1 \, . \tag{5}$$

Employ an ODE routine from GSL. Before calling the ODE routine reduce the argument to the appropriate range using the formulae

$$\sqrt{\frac{1}{x}} = \frac{1}{\sqrt{x}} \, , \; \sqrt{4x} = 2\sqrt{x} \, . \tag{6}$$

Compare with the corresponding function from `<math.h>` or from GSL.

4. **Square root (ODE representation II).** Implement a function that calculates the square root of a real positive number x using the ODE representation (check that this equation does actually has the square root as the solution, otherwise correct the equation),

$$\frac{dy}{dx} = \frac{y}{2x} \, , \; y(1) = 1 \, . \tag{7}$$

Employ an ODE routine from GSL. Before calling the ODE routine reduce the argument to the appropriate range using the formulae

$$\sqrt{\frac{1}{x}} = \frac{1}{\sqrt{x}} \, , \; \sqrt{4x} = 2\sqrt{x} \, . \tag{8}$$

Compare with the corresponding function from `<math.h>` or from GSL.

5. **Square root (root finding).** Implement a function that calculates the square root r of a real positive number x by solving the equation

$$r^2 - x = 0 \, . \tag{9}$$

Reduce the argument as in "Square root (ODE representation)" and use the initial guess of 1. Use one of the GSL's root finding routines.

Compare with the corresponding function from `<math.h>` or from GSL.

6. **Cubic root (root finding).** Implement a function that calculates the cubic root c of a real positive number x by solving the equation

$$c^3 - x = 0 \, . \tag{10}$$

Reduce the argument as in "Cubic root (ODE representation)" and use the initial guess in the middle of the interval. Use one of the GSL's root finding routines.

Compare with the corresponding function from `<math.h>` or from GSL.

7. **Cubic root (ODE representation I).** Implement a function that calculates the cubic root of a real positive number x using the ODE representation (check that this equation does actually have the cubic root as the solution, otherwise correct the equation),

$$\frac{dy}{dx} = \frac{1}{3y^2} \, , \; y(1) = 1 \, . \tag{11}$$

Employ an ODE routine from GSL. Before calling the ODE routine reduce the argument to the appropriate range using the formulae

$$\sqrt[3]{\frac{1}{x}} = \frac{1}{\sqrt[3]{x}} \, , \; \sqrt[3]{8x} = 2\sqrt[3]{x} \, . \tag{12}$$

Compare with the corresponding function from `<math.h>` or from GSL.

8. **Cubic root (ODE representation II).** Implement a function that calculates the cubic root of a real positive number x using the ODE representation (check that this equation does actually have the cubic root as the solution, otherwise correct the equation),

$$\frac{dy}{dx} = \frac{y}{3x}\,,\; y(1) = 1\,. \tag{13}$$

Employ an ODE routine from GSL. Before calling the ODE routine reduce the argument to the appropriate range using the formulae

$$\sqrt[3]{\frac{1}{x}} = \frac{1}{\sqrt[3]{x}}\,,\; \sqrt[3]{8x} = 2\sqrt[3]{x}\,. \tag{14}$$

Compare with the corresponding function from `<math.h>` or from GSL.

9. **Exponential function (ODE representation).** Implement the exponential function of a real number x using the ODE representation

$$\frac{dy}{dx} = y\,,\; y(0) = 1\,. \tag{15}$$

Reduce the argument to $0 \leq x < 1$ using the formulae

$$\exp(-x) = \frac{1}{\exp(x)}\,,\; \exp(x) = \exp\left(\frac{x}{2}\right)^2 \tag{16}$$

and employ one the GSL's ODE routines.

Compare with the corresponding function from `<math.h>` or from GSL.

10. **Sine function (ODE representation).** Implement the sine function of a real number x using the ODE representation

$$y'' = -y\,,\; y(0) = 0\,,\; y'(0) = 1\,. \tag{17}$$

Reduce the argument to, say, $0 \leq x \leq 2\pi$ using the periodicity and the symmetry of the sine function (and perhaps even further down to $0 \leq x \leq \pi/2$ using trigonometric identities) and then call one of the GSL's ODE routines.

Compare with the corresponding function from `<math.h>` or from GSL.

11. **Cosine function (ODE representation).** Implement the cosine function of a real number x using the ODE representation

$$y'' = -y\,,\; y(0) = 1\,,\; y'(0) = 0\,. \tag{18}$$

Reduce the argument to, say, $0 \leq x \leq 2\pi$ using the periodicity and the symmetry of the cosine function (and perhaps even further down to $0 \leq x \leq \pi/2$ using trigonometric identities) and then call one of the GSL's ODE routines.

Compare with the corresponding function from `<math.h>` or from GSL.

12. **Tangent function (ODE representation).** Implement the tangent function of a real number x using the ODE representation

$$y' = 1 + y^2\,,\; y(0) = 0\,. \tag{19}$$

Reduce the argument to, say, $0 \leq x < \pi/2$ using the periodicity and the symmetry of the tangent function and then call one of the GSL's ODE routines.

Compare with the corresponding function from `<math.h>` or from GSL.

13. **Implement the Arcsine function using integral representation.**

$$\arcsin(x) = \int_0^x \frac{1}{\sqrt{1-z^2}}\,dz, \qquad |x| \leq 1\,. \tag{20}$$

Compare with the corresponding function from `<math.h>` or from GSL.

14. **Implement the Arccosine function using integral representation.**

$$\arccos(x) = \int_x^1 \frac{1}{\sqrt{1-z^2}}\, dz, \qquad |x| \le 1\,. \tag{21}$$

15. **Arcsine function (root finding).** Implement the Arcsine function, $a = \arcsin(x)$, by solving the equation

$$\sin(a) = x\,. \tag{22}$$

16. **Arccosine function (root finding).** Implement the Arccosine function, $a = \arccos(x)$, by solving the equation

$$\cos(a) = x\,. \tag{23}$$

17. **Arctangent function (root finding).** Implement the Arctangent function, $a = \arctan(x)$, by solving the equation

$$\tan(a) = x\,. \tag{24}$$

18. **Arccotangent function (root finding).** Implement the Arccotangent function, $a = \operatorname{arccot}(x)$, by solving the equation

$$\cot(a) = x\,. \tag{25}$$

19. **Implement the Arctangent function using integral representation.**

$$\arctan(x) = \int_0^x \frac{1}{z^2+1}\, dz\ . \tag{26}$$

To facilitate numerical integration reduce the argument to a reasonable interval (e.g. $[0,1]$) using the formulae (check them),

$$\arctan(-x) = -\arctan(x) \tag{27}$$

$$\arctan\left(\frac{1}{x}\right) = \frac{\pi}{2} - \arctan(x)\,,\ \text{if } x > 0\,. \tag{28}$$

prior to integration. Compare with the corresponding function from `<math.h>` or from GSL.

20. **Implement the Arccotangent function using integral representation.**

$$\operatorname{arccot}(x) = \int_x^\infty \frac{1}{z^2+1}\, dz\ . \tag{29}$$

To facilitate numerical integration reduce the argument to a reasonable interval (e.g. $[1,\infty[$) using the formulae (check them),

$$\operatorname{arccot}(-x) = \pi - \operatorname{arccot}(x) \tag{30}$$

$$\operatorname{arccot}\left(\frac{1}{x}\right) = \frac{\pi}{2} - \operatorname{arccot}(x)\,,\ \text{if } x > 0\,. \tag{31}$$

prior to integration. Compare with the corresponding function from `<math.h>` or from GSL.

21. **Arctangent and Arccotangent (using complementarity and mutual recursion).** Implement the two functions,

$$\arctan(x) = \int_0^x \frac{1}{z^2+1}\, dz\ , \tag{32}$$

and

$$\operatorname{arccot}(x) = \int_x^\infty \frac{1}{z^2+1}\, dz\ , \tag{33}$$

by first reducing the argument to $x > 0$ and then, using the complementarity relation

$$\operatorname{arccot}(x) + \arctan(x) = \frac{\pi}{2}\,, \tag{34}$$

employ either the expression for $\arctan(x)$, if $x \le 1$, or $\operatorname{arccot}(x)$, if $x > 1$. See our the error-function example for inspiration. (Optional) See whether $x = 1$ is the optimal point to switch the integration routine.

22. **Implement the Arctangent function using ODE representation.**

$$\arctan(x)' = \frac{1}{x^2+1}\,,\ \arctan(0) = 0\,. \tag{35}$$

Compare with the corresponding function from `<math.h>` or from GSL.

23. **Implement the Arccotangent function using ODE representation.**

$$\text{arccot}(x)' = \frac{-1}{x^2+1}\,,\ \text{arccot}(0) = \frac{\pi}{2}\,. \tag{36}$$

Compare with the corresponding function from `<math.h>` or from GSL.

24. **Implement the Bessel function of the first kind of integer index using Bessel's integral representation.**

$$J_n(x) = \frac{1}{\pi}\int_0^\pi \cos\left(nt - x\sin(t)\right) dt\,. \tag{37}$$

Compare with the corresponding function from `<math.h>` or from GSL.

25. **Implement the Logistic function using ODE representation.**

$$y' = (1-y)y\,,\ y(0) = \frac{1}{2}\,. \tag{38}$$

26. **Diagonalisation of random real symmetric matrices.** Implement a function that takes a positive integer number n as the argument, generates an $n \times n$ random real symmetric matrix (with random elements uniformly distributed in [0,1]), diagonalizes it, and returns the largest eigenvalue as the result. Make a plot of the function (e.g. for $1 \le n \le 50$) and a reasonable fit. You can generate (pseudo)random real numbers uniformly distributed between zero and one using the `rand` function from `stdlib`,

```
#define RND (double)rand()/RAND_MAX
```

You can sort eigenvalues with `gsl_sort_vector` function.

27. **Diagonalization of projection matrices.** Implement a function that takes a positive integer number n as the argument, generates an $n \times n$ real symmetric matrix where all matrix elements are equal one, diagonalizes the matrix, and returns the largest eigenvalue. Make a plot of the function and, optionally, explain the result. You can sort eigenvalues with `gsl_sort_vector` function.

28. **Small matrix diagonalisation I.** Diagonalise numerically the matrix

$$\begin{pmatrix} x & 1 \\ 1 & -x \end{pmatrix} \tag{39}$$

and plot the eigenvalues as function of x.

29. **Small matrix diagonalisation II.** Diagonalise numerically the matrix

$$\begin{pmatrix} 1 & x \\ x & -1 \end{pmatrix} \tag{40}$$

and plot the eigenvalues as function of x.

30. **Minimisation ($\hbar\omega$).** Find numerically the minimum of the function

$$f(x) = \frac{1}{4}\left(x + \frac{1}{x}\right) \tag{41}$$

for $x > 0$. Plot the function and indicate the found minimum.

31. **Minimisation (H).** Find numerically the minimum of the function

$$f(x) = \frac{1}{2}\left(x^2 - \frac{1}{2}x\right) \tag{42}$$

for $x > 0$. Plot the function and indicate the found minimum.

32. **Exponential Integral.** Implement the 'exponential integral' function $E_1(x)$ using the integral representation

$$E_1(x) = \int_x^\infty \frac{e^{-t}}{t} dt \; . \tag{43}$$

Compare with `gsl_sf_expint_E1`.

33. **Elliptic Integral of First Kind.** Implement the 'elliptic intergal of first kind' function $F(\phi, k)$ using the integral representation

$$F(\phi, k) = \int_0^\phi dt \frac{1}{\sqrt{1 - k^2 \sin^2(t)}} \; . \tag{44}$$

Compare with `gsl_sf_ellint_F`.

34. **Elliptic Integral of Second Kind.** Implement the 'elliptic intergal of second kind' function $E(\phi, k)$ using the integral representation

$$E(\phi, k) = \int_0^\phi dt \sqrt{1 - k^2 \sin^2(t)} \; . \tag{45}$$

Compare with `gsl_sf_ellint_E`.