Exercises for 'Meteorologische Modellierung' part 'Numerical Solutions'
Semester 1: Basics

0) Preliminaries and FORTRAN

0a) The linear decay (evolution) equation

Summary
To exercise methods for the discretization in time we consider the linear differential equation

$$\frac{dT}{dt} = -aT + F \qquad \textit{(decay equation; ger: Evolutionsgleichung)}$$

This equation represents the temporal evolution of a quantity T caused by a constant forcing F and a linear damping with time scale $\tau = 1/a$. This problem can be solved analytically: $T(t)=F/a+\Delta T\ exp(-at)$. With F/a being the equilibrium (stationary) solution and ΔT the deviation of the initial value T_0 $(=T(t_0))$ from the stationary solution. We will use the analytic solution to evaluate numerical solutions.

Task:
Write a FORTRAN program which computes and writes a time series of the analytic solution of the decay equation (i.e. $T(t)$) for a given set of parameter: F=1., a=0.05, and T_0=0. The time series $T(t)$ should be computed and written for t=0, Δt, $2\Delta t$,...,$200\Delta t$ with Δt =1. Plot the time series (with, e.g., gnuplot).

0b) The linear advection equation

Summary
Advective processes play a major role in fluid dynamics. Thus, numerical schemes for discretization in time and space are introduced utilizing the (one-dimensional) linear advection equation:

$$\frac{\partial T}{\partial t} = -u\frac{\partial T}{\partial x}$$

with constant (in space and time) velocity u. The analytic solution of this equation is given by $T(x,t)=f(x-ut)$, with any function f (with argument$(x-ut)$). In our case, f should be given by a super-position of waves with wavenumber k (and phase velocity u):

$$T(x,t) = \sum_k \mathrm{Re}\left(T_k \exp(ik(x-ut))\right)$$

Task:
Write a FORTRAN program, which computes $T(x,t)$ for a traveling wave with wave number one. Use NX (also x=0, Δx, $2\Delta x$,...,$NX\Delta$x) grid points to represent the wave in space. Use the following parameter: NX=100, Δx=1., u=10. The amplitude of the wave (T_1) should be 10. he program should compute and write (i) the time series $T(x,t)$ for x=0 and t=0, Δt, $2\Delta t$,...,$200\Delta t$ with Δt =1, and (ii) the spatial distribution $T(x,t)$ for t=0 und t=$199\Delta t$ (all grid points x). Plot the time series and the spatial distribution (e.g. with gnuplot).

Extension: Write the complete wave (all grid points) at every time step, and plot a Hovmöller diagram of the wave (e.g. with grads).

1) Discretization in time: The decay (evolution) equation

Tasks:

1. Solve the decay equation numerically. Include the numerical solution into your FORTRAN program for the analytic solution (task 0a). Use the same parameter as for the analytic solution. Use the explicit Euler scheme with time step $\Delta t=1$. Write the time series of the numerical solution ($t=0,...200\Delta t$) and plot it together with the analytic solution.

2. Based on experiments (by changing the time step and utilizing the RMS error and the maximum norm), identify the order (error) of the explicit Euler scheme

3. Use, in addition to 1, the implicit, the Crank-Nicolson, the Runge-Kutta, and the Adams-Bashforth scheme (same parameter as in 1). Compare the accuracy of the different schemes with same time step by computing the error (RMS and maximum), and identify (again by experiment) the order of the schemes.

Additional task (optional):

The numerical schemes for the discretization in time can, of course, also be used to solve more complex problems. A well-known dynamical system is the Lorenz model, which may be seen, for example, as a low order system for atmospheric convection. The Lorenz model is the prototype model for deterministic chaos. The equations are given by

$$\frac{\partial X}{\partial t} = -\sigma X + \sigma Y$$

$$\frac{\partial Y}{\partial t} = -XZ + rX - Y$$

$$\frac{\partial Z}{\partial t} = XY - bZ$$

with external parameters σ (Prandtl number), r (Rayleigh number), and b (damping).

Compute the numerical solution of the Lorenz model and plot the famous 'butterfly' (X(t),Y(t),Z(t)). Classical values of the parameters are σ =10, r=28, and b=8/3. Initial condition may be (1,1,1). An appropriate time step is 1/100., and approx. 10000 time steps should be computed. Note: since the Lorenz model is very sensitive, real scientific studies should be conducted by using a highly accurate scheme like the Runge-Kutta. However, here you may first try the explicit Euler. All variables in the program should be defined as high precision (kind=8).

2) Discretization in time and space: The linear advection equation

Tasks:

1. Compute and plot the numerical solution of the wave advection (see task 0b). Use cyclic boundary conditions. Utilize the grid point methods 'upstream' and 'Leapfrag' (with and without Asselin filter). Attention(!): Check the numeric stability and, if needed, adopt the time step Δt.

2. Assess (experimentally) the error in amplitude and in phase of both (upstream and Leapfrog) schemes in your setup for task 1 for wave numbers 1, 2, and 10.

3. Compute and plot the numerical solution of the advection of a 'box' (T(t=0) = 10 for x <= 50, T(t=0) = 0 otherwise). Use the same setup as in tasks 1 and 2.

Additional task (optional)

Use the semi-Lagrange scheme (backward trajectories with linear interpolation) to solve the linear advection equation (setup as in tasks 1-3).

3) Discretization in time and space: Diffusion and transport equation

Task:

Expand your program solving the advection equation to solve the transport equation. Use the time splitting method utilizing Leapfrog for advection and FTCS for diffusion. Use the same setup as in the advection case plus a diffusion coefficient of 2.

4) Discretization in time and space: non-linear advection and transport (Burgers equation)

Task:

Try to solve (numerically) the viscid and inviscid Burgers equation. Compute the time evolution of a simple sin-wave with wave number 1 (see linear advection).