Exercise 4 – 2D diffusion

Rewrite your Fortran program from Exercise 3, Task 3 such that it solves the two-dimensional heat diffusion equation:

$$\frac{\partial T}{\partial t} = \kappa \cdot \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) = \kappa \nabla^2 T \tag{1}$$

1. Read in the control parameters from a namelist:
   - Number of grid points in x and y direction `nx`, `ny`.
   - Diffusion coefficient `kappa` (start with `kappa=1`)
   - Simulation time `total_time` (start with `total_time=0.1`)
   - Time step constant `a`.
2. Initialize the variables
   - Temperature field: 2D array of size `(nx,ny)` with random numbers or delta function (spike)
   - Grid spacing `h=1./(ny-1.)`
   - Time step `dt=a*h**2/kappa`
   - Number of time steps `nsteps=total_time/dt`
3. Write a loop that executes `nsteps` time steps.
   - Calculate the second derivative with a 2D version of your subroutine from Exercise 2:

$$\nabla^2 T_{i,j} \approx \frac{T^n_{i+1,j} + T^n_{i-1,j} + T^n_{i,j+1} + T^n_{i,j-1} - 4 \cdot T^n_{i,j}}{h^2} \tag{2}$$

   - Integrate forward in time:

$$T^{n+1}_{i,j} = T^n_{i,j} + \Delta t \cdot \kappa \cdot \nabla^2 T_{i,j} \tag{3}$$

   - Use the boundary conditions `T=0` (at the beginning as well as after each time step).
   - Write the temperature field at the first and last time step (or more) into a text file using the `OPEN()` and `WRITE()` statements.
4. Run your program and test it.
   - Use two different initial conditions for the temperature (`random` and `spike`) and vary the number of grid points.
   - Determine the critical value for `a` above which the solution becomes unstable.
   - Plot the resulting temperature fields with your favorite plotting tool.
   - Optional: If you write out multiple time steps you can create an animated gif as follows:

```
micromamba activate /headless/envs/magic
convert -delay 20 *.png animation.gif
```

Deadline: Please hand in your solutions (`.f90` files and plots) until **Tuesday, 16 April, 23:59**.