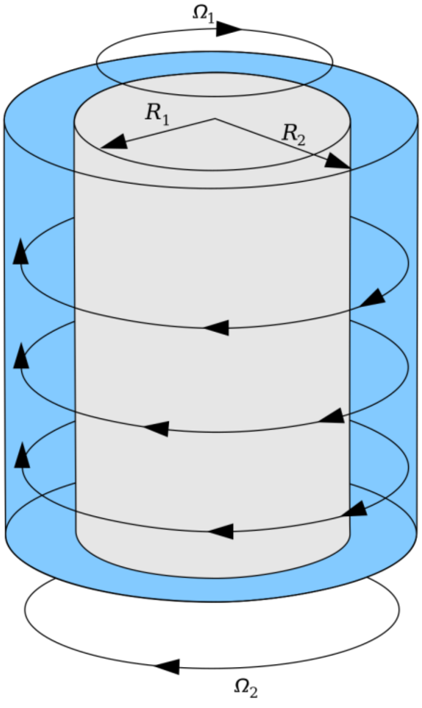
8c. Couette Flow

couette.cpp: A fluid fills the gap between an inner cylinder (with radius r_1) rotating at angular velocity Ω_1 and an outer cylinder (with radius r_2) rotating with angular velocity Ω_2.



The flow angular velocity $u = \Omega r$ obeys the following ODE:

$$\frac{d^2u}{dr^2} + \frac{1}{r}\frac{du}{dr} - \frac{u}{r^2} = 0$$

where $u_1 = 1$ at the inner cylinder ($r = r_1 = 1$) and $u_2 = 0.6$ at the outer cylinder ($r = r_2 = 10$).

- Solve the problem using two different algorithms:
  1. Shooting method, by integrating the previous ODE with the 4th order Runge-Kutta algorithm and `NSTEPS = 200` (*Hint*: $u'(r_1) < 0$). Use a root-finder of your choice (`xtol=1.e-8`). Report, in the comments at the beginning of the C++ code, the value of $u'(r_1)$ that you obtain.

  2. Finite difference method, with a grid of `(NSTEPS+1)` points (inclusive of boundary values). *Hint*: write the tridiagonal system resulting from a finite difference discretization of the 2nd and 1st derivatives and obtain the coefficients `a[]`, `b[]`, `c[]` and `r[]` by imposing the correct boundary conditions.

- Using the analytical solution:

$$u^{ex} = \frac{a}{r} + br \quad \text{where} \quad a = \frac{r_1 r_2 (r_1 u_2 - r_2 u_1)}{r_1^2 - r_2^2}, b = \frac{u_1 r_1 - u_2 r_2}{r_1^2 - r_2^2}$$

Compute the L1 norm errors* for the two methods.

---

* $\epsilon = \frac{1}{N}\sum |u_i - u_i^{ex}|$

8c. Couette Flow (cont)

- Upload your code with your last name and the output inserted in the comment at the beginning of the file and the *necessary* library functions at the end:

```
// Last name: ...
// u'(r1)        = ...
// L1 err(Shooting)     = ...
// L1 err(Finite Diff)  = ...
#include ...
...
int main()
{
  // code here
}

void RK4Step (...){
...
}

void Residual (...){
...
}

void RHS (...){
...
}
```

- Also, upload a png (or jpeg or pdf) plot showing the solution u(r) obtained with one of the two methods.