Computational Continuum Modelling
Practical 4: FORCE for the Euler equations

We now take your FORCE scheme for Burgers' equation (and the advection equation) and use this to solve the Euler equations. This will modifications to the data structures within your code, but the algorithms shouldn't have to change.

It is recommended that your data structure stores the conserved variables, since these are the ones that are updated by

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n - \frac{\Delta t}{\Delta x}\left(\mathbf{f}_{i+1/2}^n - \mathbf{f}_{i-1/2}^n\right).$$

However, you will find it useful to implement functions to convert conserved variables to primitive, and primitive to conserved. This needs to happen fairly often, in computing pressure for the fluxes and in computing the sound speed. It is also common to give initial data, and to provide output, in terms of the primitive variables.

For all tests with the Euler equations, transmissive boundaries are a sensible choice. For these equations, the increase in computational effort might mean that simulations take a little bit longer to run, especially if you are using a lot of cells (several thousand). At the same time, as code gets more complex, bugs can cause unexpected effects, such as the time step approaching zero. Ensure that your code produces regular output to screen whilst running, displaying e.g. time and time step size. Investigate how compiling with optimisation turned on or off affects run times.

Exercises:

There are five key tests to simulate for the Euler equations, given below, and also in Table 4.1 in Toro's book (note that Toro's convention is to call velocity u). This table gives the initial data for the primitive variables across the computational domain, in each case you are setting up a Riemann problem.

Test	ρ_L	v_L	p_L	ρ_R	v_R	p_R	time
1	1.0	0.0	1.0	0.125	0.0	0.1	0.25
2	1.0	-2.0	0.4	1.0	2.0	0.4	0.15
3	1.0	0.0	1000.0	1.0	0.0	0.01	0.012
4	1.0	0.0	0.01	1.0	0.0	100.0	0.035
5	5.99924	19.5975	460.894	5.99242	-6.19633	46.0950	0.035

Each test should be run on a domain $x \in [0, 1]$, where the subscript $_L$ is the initial data for $x \leq 0.5$ and the subscript $_R$ is the initial data for $x > 0.5$, and the final time for the simulation is given. In each case, a $\gamma = 1.4$ ideal gas equation of state should be assumed, and a CFL number of 0.8 is recommended.

Section 4 of Toro's book, directly after the table, gives exact solutions for these tests. Some of these tests may require a lot of computational cells for the FORCE scheme to give comparable looking results, and even then, some features may never look quite correct. Try running the tests with varying numbers of cells, it is common to see 100, 200, 400 and 800 cells used for tests, but also try a very high number to see how things improve.

If you want to make sure results are correct (without writing an exact solver for these Riemann problems), Table 4.3 in Toro's book gives the numerical values for the intermediate waves you see in your solution.