MTH 654-2024 Homework B. **FirstName LastName**
Date: November 22, 2024
Prof. M. Peszynska

Please turn in 2 pbms.

Instructions: Your answers should be concise but also provide sufficient insight into the algorithms and theory rather than be "*a bunch of plots* stapled together". Avoid showing `obvious code snippets` and ELEMENTARY calculations. Code in any language, but using MATLAB might save you time, and allow Instructor's support.
Items to be addressed in blue.

Problem 1 (+10): Reality check on data and modeling error. (Choose two of a-b-c)).
(a, +5) Consider the data $k = 10^{-12}[m^2]$, $\mu = 10^{-3}[Pa \cdot s]$ for incompressible Darcy flow due to pressure boundary conditions and some $\Delta p = p_{right} - p_{left}$. Answer: What Δp must be applied for the flow to have velocity (i) $1[mm/day]$, (ii) $1[mm/s]$, (iii) $1[m/s]$?
To see which (if any) is realistic, compare Δp to the difference of pressures ΔP_{grav} which balances the gravity $\rho G \Delta D$ for $\Delta D = 1[m]$, with $\rho \approx 10^3[kg/m^3]$, and $G \approx 10[m/s^2]$.
(b, +5) Consider the flow field $q(x, y)$ given below, and project q to the field Q_h on a $M_x \times M_y$ grid over $(-1, 1)^2$. Answer: Is q divergence free? Is Q_h also discrete-divergence free ? (Note $Q_h \in V_h$, the discrete space of velocities from Raviart-Thomas space on quadrangles obtained when solving elliptic PDE with FV/CCFD).

$$q(x, y) = (1, 2); \;\; q(x, y) = (y, -x); \;\; q(x, y) = (x, y); \tag{1a}$$

$$q(x, y) = (x^2y, -xy^2); \;\; q(x, y) = (x^2y + e^y, -xy^2 + e^x) \tag{1b}$$

If possible, construct your own $q(x, y)$ for which $\nabla \cdot q = 0$, but $\nabla_h \cdot Q_h \neq 0$.
(c, +5) Following the class discussion, consider the transport model $(cu)_t + (qu)_x = 0$. Let c be found from some coupled model, so that in practice you work with $\tilde{c}$ rather than with c. Estimate the modeling error depending on $c - \tilde{c}$. What if also q is replaced by $\tilde{q}$.

Solution:

Problem 2 (+10); TRANSPORT in 2D. Implement the transport model to simulate

$$\partial_t(cu) + \nabla \cdot (qu) = 0, \; x \in \Omega, 0 < t \leq T_{end}; u(x, 0) = u_{init}(x); \; u|_{\Gamma_{in}} = 0. \tag{2}$$

You can start by extending the code given in `TRANSPORT1d.m` to the grid $M_x \times M_y$.
Hint: (*Do not turn in unless issues arise:*) Test first with $c \equiv 1$ and some discrete-divergence free field on a coarse grid, then with some other field. Your code should handle the inflow boundary conditions appropriately. Contact me if you need help getting started.
Example to turn in: Produce the numerical solutions for the field Q_h, a projection of i) $q(x, y) = (-1, -1)$, (ii) $q(x, y) = (y, -x)$. Let $c \equiv 1$.
For **extra credit**, *choose also something fancy such as* $c|_{\Omega_{lrcorner}} = 0.5$ *where* $\Omega_{lrcorner} = \{(x, y) \in \Omega : \text{dist}((x, y), (1, -1)) \leq 1\}$.
Let U_{init} be a projection of $u_{init}(x) = \chi_{\Omega_{blob}}(x)$, where Ω_{blob} is located by the following code

```
midpointx = floor([nx/3*2,nx/6*5]);midpointy=floor([ny/3*2,ny/6*5]);
unit=zeros(nx,ny,1); uinit(midpointx(1):midpointx(2),midpointy(1):midpointy(2))=1;
```

Please **plot your solutions** for (i), (ii) at $t = 1, t = 2, t = 3$ for grid 50x50, 100x50, and 100x100. (**Hint:** My solutions are plotted in Figure 1).

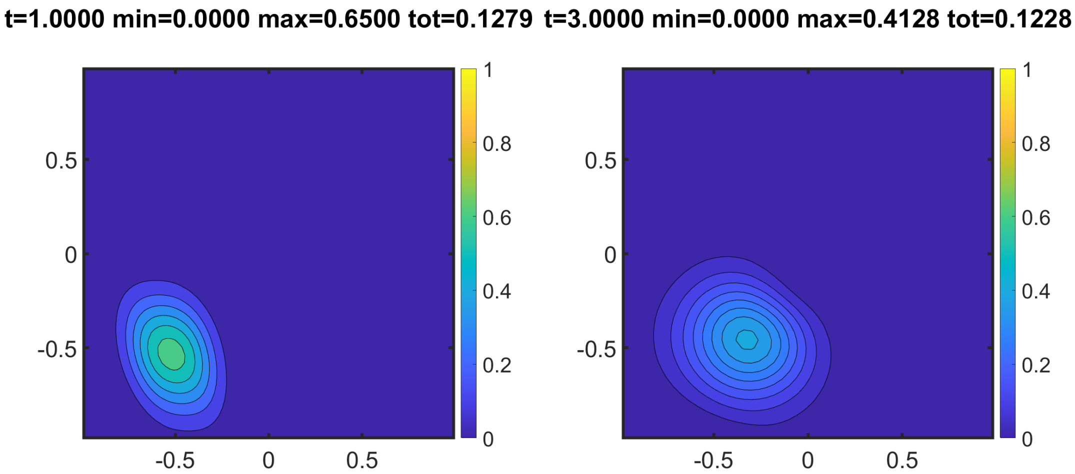

FIGURE 1. Approximate solutions to (2) at $t = T_{end}$ as shown. Left: velocity (i) and grid 100x50. Right: velocity (ii), and 100x100.

Answer the following questions based on your implementation:

(1) Case (i). At what time t does your approximation to $u(x, t)$ essentially disappear for grid 50x50? What about for grid 100x50? 100x100?

(2) Case (ii). At what time t does your approximation to the total amount $\int_\Omega u(x, t)dx$ start to differ significantly (more than 10^{-4}) from $\int_\Omega u_{init}(x)dx$ for grid 50x50? What about for grid 100x100?

Based on these answers, do you think the Godunov scheme in 2D has the properties you would desire? (What would you expect from the true solution?)

Extra credit: write your own FLOW code to get the flux solution Q_h for some attractive looking boundary conditions and heterogeneous k.

Solution: