**Option 1**

Build a discontinuous Galerkin solver for the first order advection equation. The solver must do the following:

i. Read a .neu triangle mesh file (as in the finite volume solver).

ii. For each edge of each triangle find the neighbor edge, if any.

iii. Read a data file provided by TW, which includes information about the electrostatic interpolation nodes of Hesthaven on the triangle e.g. load um_n5.mat

iv. The node coordinates will be stored in umR and umS. Find:
   a. The indices of nodes on edge 1 (i.e. n such that umS(n)+1==0)
   b. The indices of nodes on edge 2 (i.e. n such that umR(n)+umS(n)==0)
   c. The indices of nodes on edge 3 (i.e. n such that 1+umR(n)==0)

   You may well have to perform the tests to some finite tolerance.

v. Compute the physical location of the nodes for each triangle, understood as the image of the node locations on each reference triangle under the following map:

\[
\begin{align*}
\left( x_n^k, y_n^k \right) &= \left( \frac{r_n + s_n}{2} x_{x,1}^k + \frac{1 + r_n}{2} x_{x,2}^k + \frac{1 + s_n}{2} x_{x,3}^k, x_{y,1}^k + x_{y,2}^k + x_{y,3}^k \right) \\
\end{align*}
\]

vi. For each node on each edge of each triangle, find which node of any neighboring triangle edge is at the same location as this node.

vii. Starting with an upwind flux DG scheme for advection:

\[
\begin{align*}
\left( v \frac{\partial u}{\partial r} \right)_{x^k} &= \left( v \frac{\partial u}{\partial x} \right)_{x^k} + \sum_{e=1}^{c} \left( \frac{n_{x}^{k,e} + p_{x}^{k,e}}{2} \right) \left( v \left[ u \right] \right)_{x^k} \\
\end{align*}
\]

we choose to represent the polynomial space with Lagrangian polynomials defined in terms of the electrostatic node sets and the above statement is discretized into:
For the k'th triangle find \( u_{k,n} \in \mathbb{R}^N \) such that

\[
(l_n, l_m)_T \frac{du_{k,m}}{dt} = \left( l_n, \frac{\partial l_m}{\partial x} \right)_T u_{k,m} + \sum_{e=1}^{e_3} \left( \frac{n^{k,e}_x + n^{k,e}_s}{2} \right) \left( l_n, l_m \right)_{T_T} \left[ u_{k,m} \right]
\]

where \( N = (p+1)(p+2)/2 \)

To simplify things we transform to the reference triangle \( T \):

For the k'th triangle find \( u_{k,n} \in \mathbb{R}^N \) such that

\[
\frac{A_k}{2} \left( l_n, l_m \right)_T \frac{du_{k,m}}{dt} = \frac{A_k}{2} \left( \left( \frac{\partial r}{\partial x} \right)_k \left( l_n, \frac{\partial l_m}{\partial r} \right)_T + \left( \frac{\partial s}{\partial x} \right)_k \left( l_n, \frac{\partial l_m}{\partial s} \right)_T \right) u_{k,m} + \sum_{e=1}^{e_3} \left( \frac{n^{k,e}_x + n^{k,e}_s}{2} \right) L_{k,e} \left( l_n, l_m \right)_{T_T} \left[ u_{k,m} \right]
\]

with summation implied on the \( m \) indices from 1 to \( N \)

Tidying up:

For the k'th triangle find \( u_{k,n} \in \mathbb{R}^N \) such that

\[
(l_n, l_m)_T \frac{du_{k,m}}{dt} = \left( \frac{\partial r}{\partial x} \right)_k \left( l_n, \frac{\partial l_m}{\partial r} \right)_T + \left( \frac{\partial s}{\partial x} \right)_k \left( l_n, \frac{\partial l_m}{\partial s} \right)_T \right) u_{k,m} + \sum_{e=1}^{e_3} \left( \frac{n^{k,e}_x + n^{k,e}_s}{2} \right) L_{k,e} \left( l_n, l_m \right)_{T_T} \left[ u_{k,m} \right]
\]

with summation implied on the \( m \) indices

Next introduce some matrix notation:

\[
\mathbf{M}_{nm} = \left( l_n, l_m \right)_T, \quad \mathbf{\hat{D}}_{nm} = \left( l_n, \frac{\partial l_m}{\partial r} \right)_T, \quad \mathbf{\hat{D}}^{s}_{nm} = \left( l_n, \frac{\partial l_m}{\partial s} \right)_T, \quad \mathbf{\hat{F}}_{nm} = \left( l_n, l_m \right)_{T_T} \left[ u_{k,m} \right]
\]

Then the statement becomes:

For the k'th triangle find \( u_{k,n} \in \mathbb{R}^N \) such that

\[
\mathbf{M}_{nm} \frac{du_{k,m}}{dt} = \left( \frac{\partial r}{\partial x} \right)_k \mathbf{\hat{D}}_{nm} + \left( \frac{\partial s}{\partial x} \right)_k \mathbf{\hat{D}}^{s}_{nm} u_{k,m} + \sum_{e=1}^{e_3} \left( \frac{n^{k,e}_x + n^{k,e}_s}{2} \right) L_{k,e} \mathbf{\hat{F}}_{nm} \left[ u_{k,m} \right]
\]

with summation implied on the \( m \) indices

Finally multiplying both sides by the inverse of \( \mathbf{M} \):
For the k'th triangle find $u_{k,m} \in \mathbb{R}^N$ such that

$$
\frac{du_{k,m}}{dt} = \left( \frac{\partial r}{\partial x}_k \right) D'_{nm} + \left( \frac{\partial s}{\partial x}_k \right) D'_{nm} \left\{ u_{k,m} + \sum_{e=1}^{e_3} F^e_{nm} \left( n^e_{x} + n^e_{y} \right) \frac{L_{e} u_{k,m}}{2} \right\}
$$

where $F^e_{nm} = (M)^{-1}_{ni} \hat{F}^e_{ni}$, $D'_{nm} = (M)^{-1}_{ni} \hat{D}^e_{ni}$ and $D_{nm}^e = (M)^{-1}_{ni} \hat{D}_{ni}^e$ with summation implied over i.

This method simply requires the evaluation of three matrix-vector products on each triangle, five scaling multiplies and the ability to construct the difference $[u_{k,m}]$ which can be evaluated as:

$$
[u_{k,m}] = \begin{cases} 
0 & \text{if } m \text{ is not an index of a node on the edge in question} \\
-u_{k,m} & \text{if this is a boundary edge} \\
u_{k,m'} - u_{k,m} & \text{otherwise and } k', m' \text{ is the neighbor node on this edge} 
\end{cases}
$$
as computed in vi.

For the project:

$N = umN$

$D' = umDr$

$D' = umDs$

$[F^1, F^2, F^3] = umLIFT$

You will also need to compute the area $A$, $dr/dx$ and $ds/dx$ for each triangle:

$$
A_k = \frac{1}{2} \left| \left( \frac{v^k_2 - v^k_1}{v^k_3 - v^k_1} \right) \left( v^k_3 - v^k_1 \right) \left( v^k_3 - v^k_1 \right) \left( v^k_3 - v^k_1 \right) \right|
$$

$$
\left( \frac{\partial r}{\partial x} \right)_k = \frac{\left( v^k_3 - v^k_1 \right)}{A_k}
$$

$$
\left( \frac{\partial s}{\partial x} \right)_k = -\left( v^k_2 - v^k_1 \right) A_k
$$

$$
\left( \frac{\partial r}{\partial y} \right)_k = -\left( v^k_3 - v^k_1 \right) A_k
$$

$$
\left( \frac{\partial s}{\partial y} \right)_k = \frac{\left( v^k_3 - v^k_1 \right)}{A_k}
$$

You will also need the edge lengths and unit outwards facing normals for each
edge of each triangle, as computed in Project 1.

viii. Use a Runge-Kutta scheme of your choosing for time stepping and solve the advection equation using the above scheme. Integrate for some small time so that the pulse stays compactly supported inside a rectangular domain and compute the order of accuracy for one of the sets of nodes using a sequence of refined meshes.

ix. Extra credit: generalize your solver to handle the linearized Euler equations we saw in Project 1. Repeat the experiment from that project with a Gaussian pulse set off inside the Irish Sea. If you have time repeat the convergence test from Project 1.

Option 2

A project of your own choosing which involves one or more of:

i. Finite difference
ii. Finite volume
iii. Discontinuous Galerkin
iv. Finite element

The project should be related to your research interests. You have 5 minutes in person or a 10 line email to pitch your idea to TW.

You must perform a validation of the order of accuracy of your method, and try to solve something non-trivial, preferably in two spatial dimensions or higher.