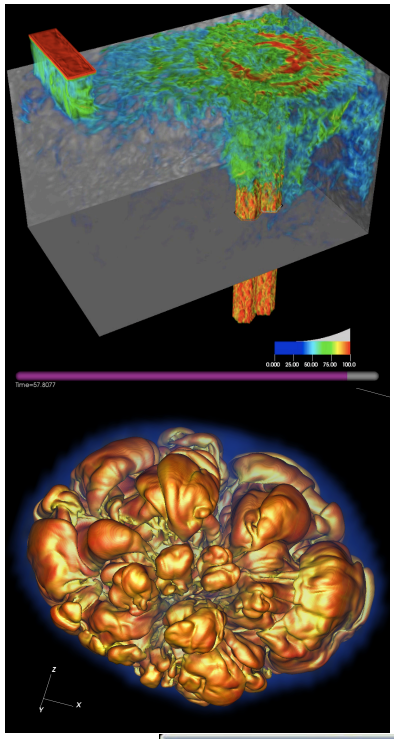
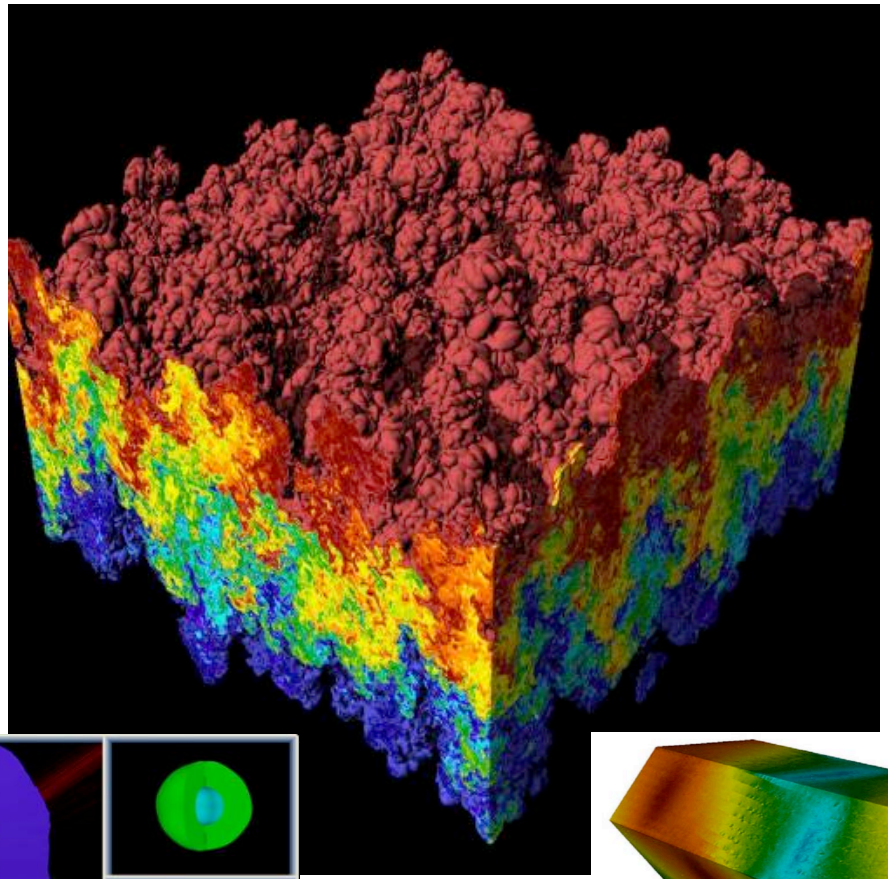
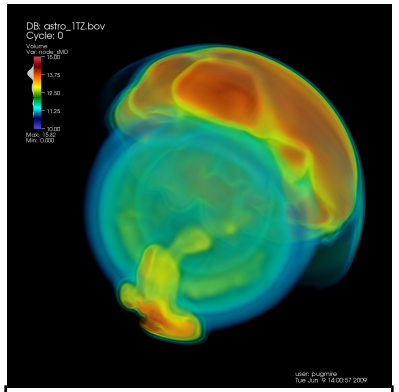
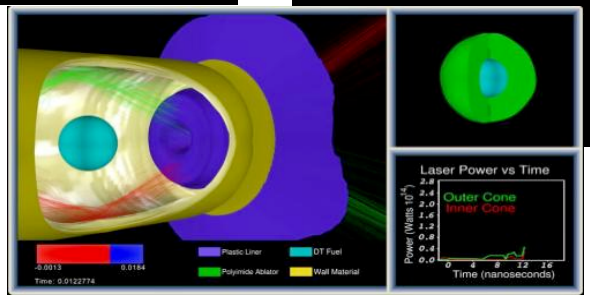
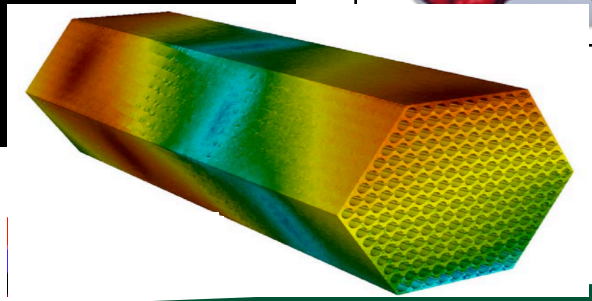
Fields, Meshes, and Interpolation (Part 2)











Hank Childs, University of Oregon

Quiz #1 on Thursday

- Worth 2 points
- Take from home or from class
- Will be on memory storage (bytes to represent things)
- Open book, open notes
- Will run for 5 minutes
  - Answers on Canvas online
  - Answers on paper in class (or Canvas)

SCENARIO	WHAT SHOULD I DO?
1. **Positive for COVID-19.** I have tested or have been confirmed positive or presumed positive by a health care provider. NOTE: *One's vaccination status is not relevant in this scenario.*	1. Stay home and isolate. Do not come to campus. Isolate for 5 days and remain masked when around others for an additional 5 days. 2. Complete the COVID-19 case and contact reporting form. 3. *Employees:* Notify your supervisor. *Students:* Notify your instructors so you can make up missed class work. **WHEN CAN I RETURN TO CAMPUS?** If you do not experience symptoms, you may return to campus 5 days after you were tested. If you experience mild or moderate symptoms, you may return to campus 5 days after your symptoms began if your symptoms are improving and you have been fever-free for at least 24 hours without the help of medication.

SCENARIO	WHAT SHOULD I DO?
2. **A vaccinated contact who has no symptoms.** I was in close contact with someone who is a COVID case (COVID-positive) during their period of transmissibility. I do not have any symptoms of COVID-19. I am up to date on my COVID-19 vaccinations.	1. Complete the COVID-19 case and contact reporting form. *Employees:* Notify your supervisor. 2. Close contacts who are up to date with their COVID-19 vaccinations do not need to quarantine but **must wear a mask around others for 10 days following your exposure.** 3. It is recommended that you test 5 days after exposure. IMPORTANT NOTE ABOUT HOUSEHOLD CONTACT: Asymptomatic vaccinated close contacts do not need to quarantine even if the person who is COVID-positive is in your home and you continue to have contact with this person. In the case of a continuous exposure, testing is recommended 3-5 days after the COVID-positive person's 5-day isolation is finished but, for your own peace of mind, you may also get tested during their isolation period.

SCENARIO	WHAT SHOULD I DO?
4. **Symptomatic but not confirmed positive for COVID-19.** I have one or more symptoms of COVID-19. These symptoms are different from my baseline. I have not consulted with a health care provider or been tested for COVID-19. *NOTE: One's vaccination status is not relevant in this scenario.*	1. Stay home. Do not come to campus 2. *Employees:* Notify your supervisor. *Students:* Notify your instructors so you can make up missed class work. 3. Consider consulting a health care provider. **WHEN CAN I RETURN TO CAMPUS?** You may return to campus only after the following conditions have been met: A. Your symptoms* are resolving and you have been fever-free for at least 24 hours AND You obtain a negative PCR or NAAT COVID-19 test. (Home tests do not meet this requirement); OR B. Your health care provider clears you to work; OR C. It has been 5 days since the onset of symptoms, your symptoms have improved, and you are fever-free.

Outline

- 6 Slide Review
- The Data We Will Study
  - ~~Overview~~
  - ~~Fields~~
  - Meshes
  - Interpolation

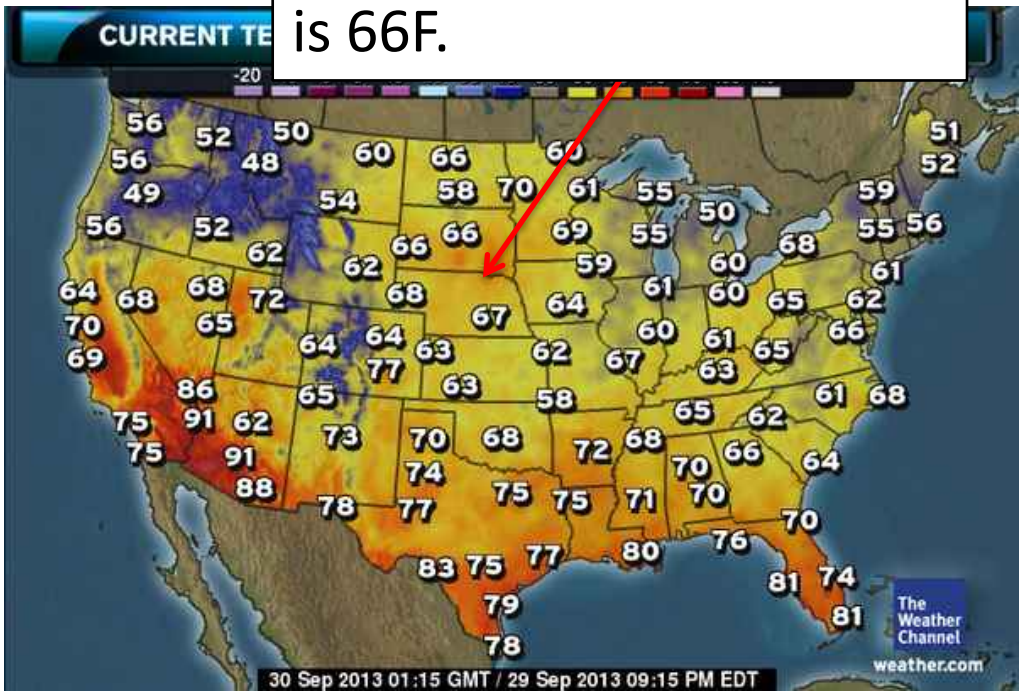
Outline

- 6 Slide Review
- The Data We Will Study
  - ~~Overview~~
  - ~~Fields~~
  - Meshes
  - Interpolation

Scalar Fields

- Defined: associate a scalar with every point in space
- What is a scalar?
  - A: a real number
- Examples:
  - Temperature
  - Density
  - Pressure

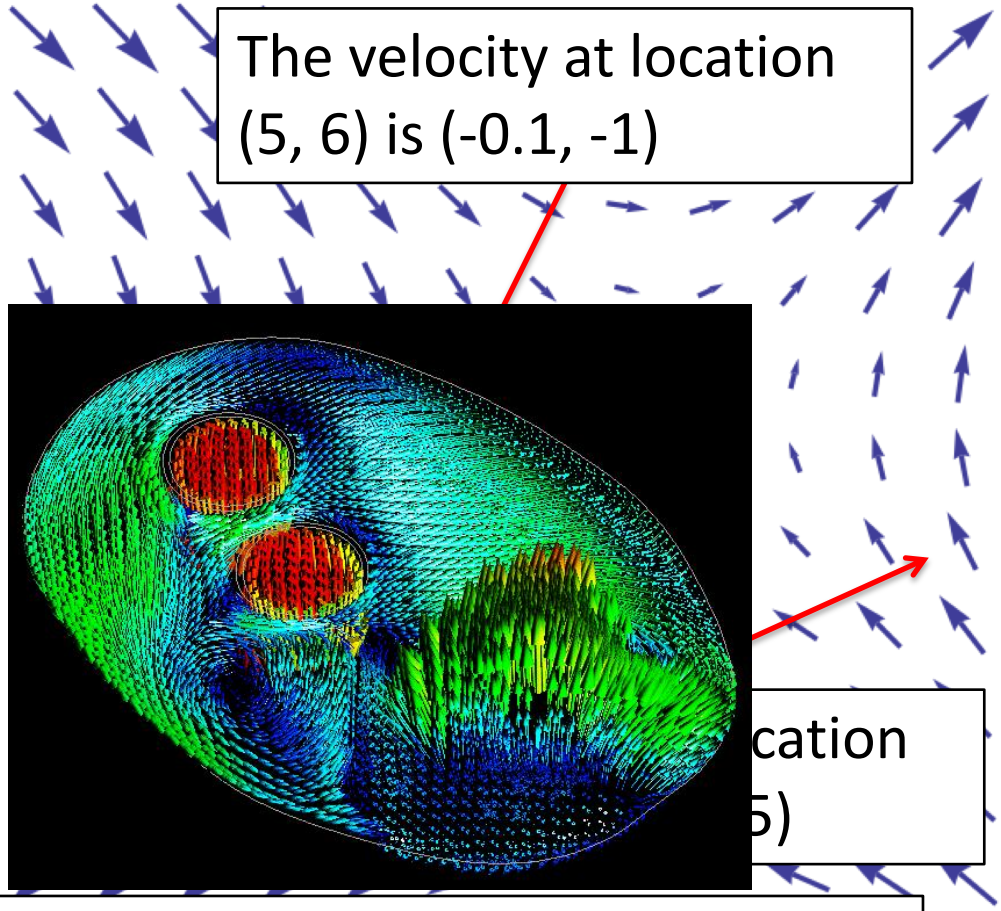
The temperature at 41.2324° N, 98.4160° W is 66F.



Fields are defined at every location in a space (example space: USA)

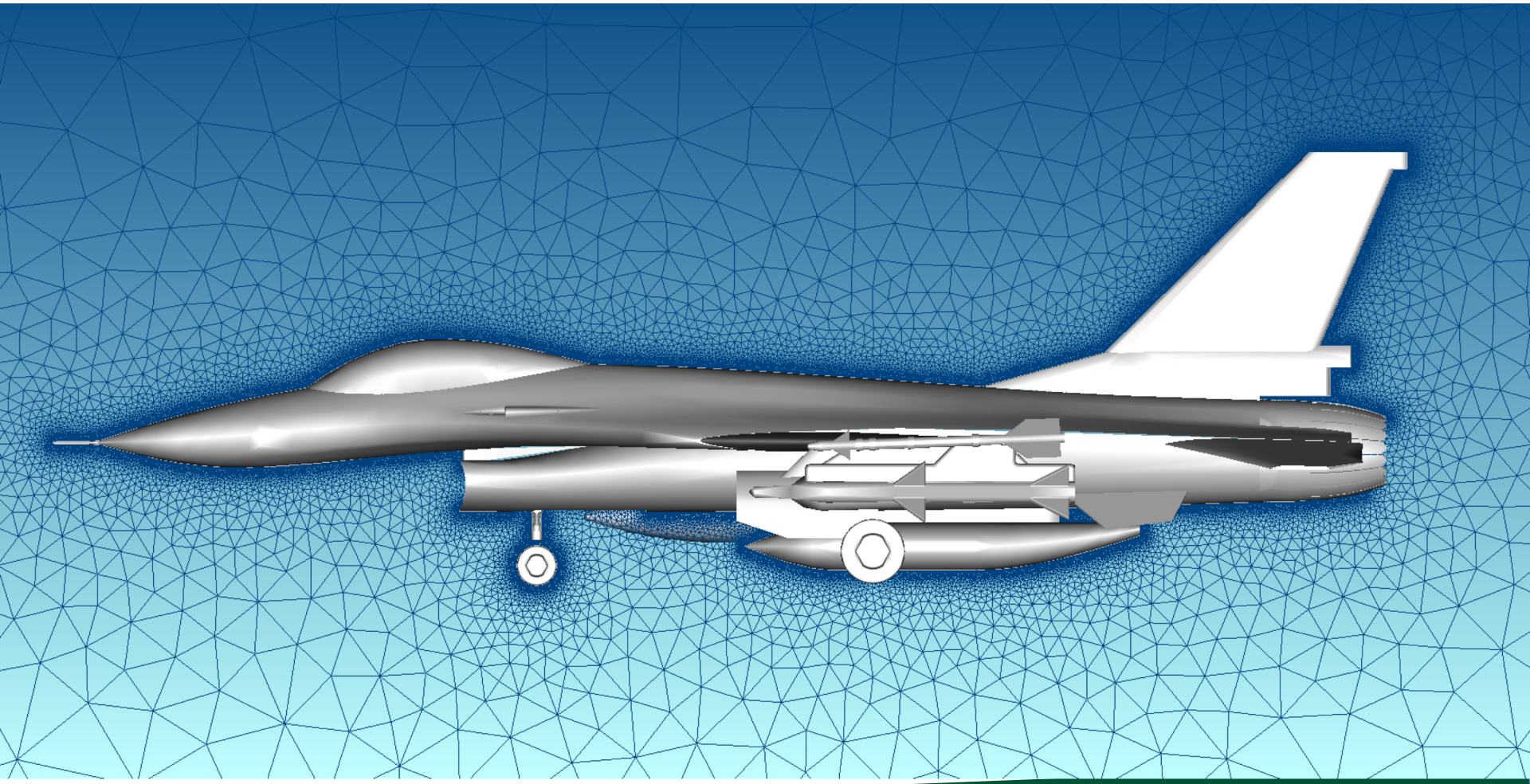
Vector Fields

- Defined: associate a vector with every point in space.
- What is a vector?
  - A: a direction and a magnitude
- Examples:
  - Velocity

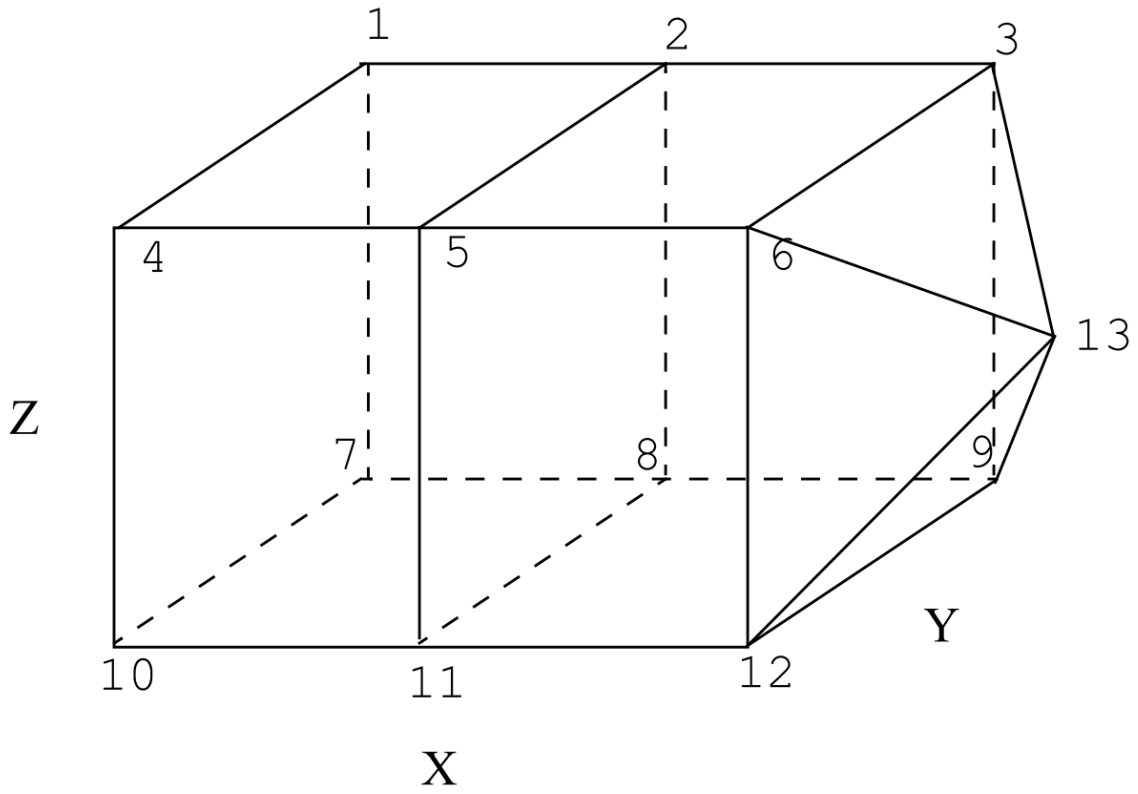


Typically, 2D spaces have 2 components in their vector field, and 3D spaces have 3 components in their vector field.

An example mesh



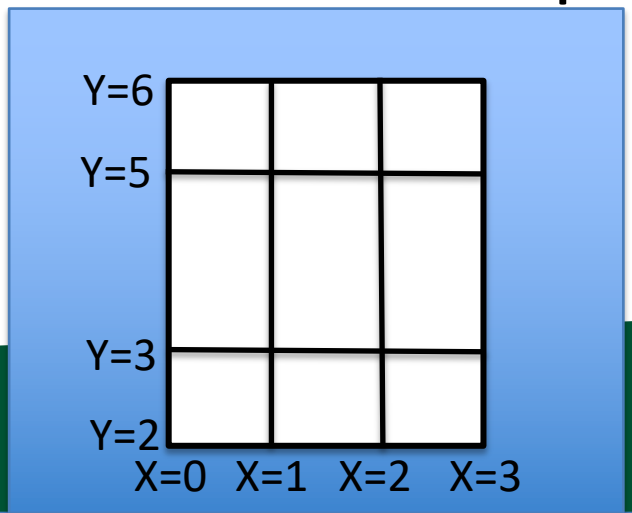
Anatomy of a computational mesh



- Meshes contain:
  - Cells
  - Vertices
- This mesh contains 3 cells and 13 vertices
- Pseudonyms:
  - Cell == Element == Zone
  - Point == Vertex == Node

Rectilinear meshes

- Rectilinear meshes are easy and compact to specify:
  - Locations of X positions
  - Locations of Y positions
  - 3D: locations of Z positions
- Then: mesh vertices are at the cross product
- Example:
  - X={0,1,2,3}
  - Y={2,3,5,6}



Definition: dimensions

- A 3D rectilinear mesh has:
  - X = {1, 3, 5, 7, 9}
  - Y = {2, 3, 5, 7, 11, 13, 17}
  - Z = {1, 2, 3, 5, 8, 13, 21, 34, 55}

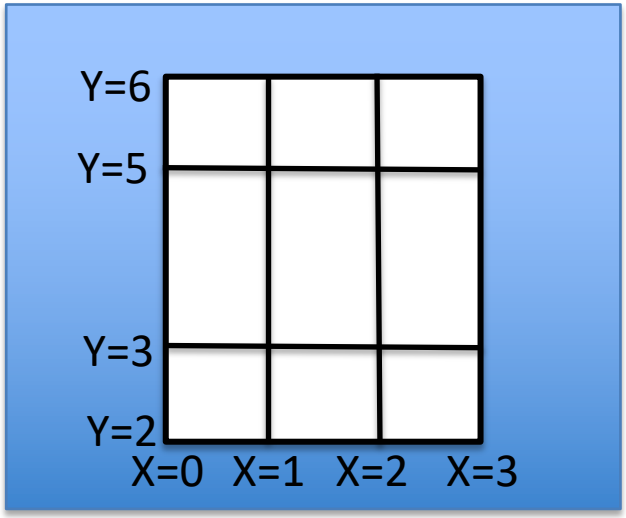
- Then its dimensions are 5x7x9

Outline

- 6 Slide Review
- The Data We Will Study
  - ~~Overview~~
  - ~~Fields~~
  - Meshes
  - Interpolation

- We are beginning to understand meshes ...
- ... but how do we represent them in memory?

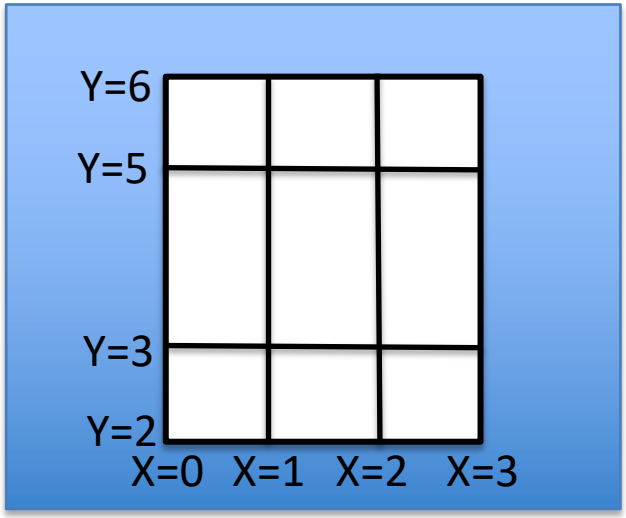
How can we represent this mesh?



(let's make a C++ class and use floats to store data)

What if there is a field on the mesh?

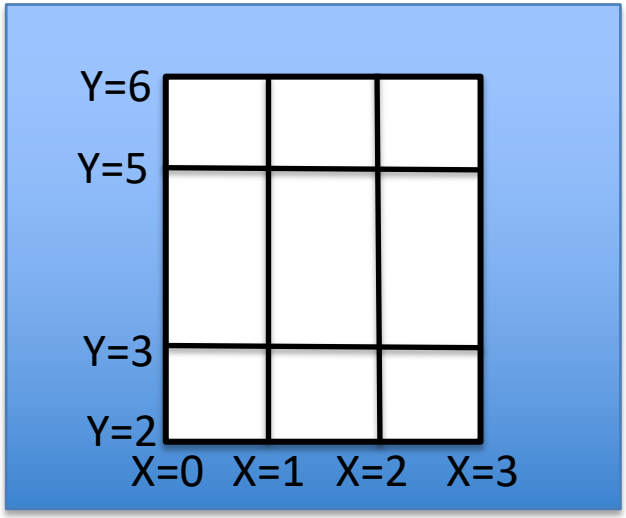
→ assume there is a temperature value at each vertex



(let's extend our C++ class)

What if there are two fields on the mesh?

→ now assume there is a temperature value and a pressure value at each vertex



(let's extend our C++ class again)

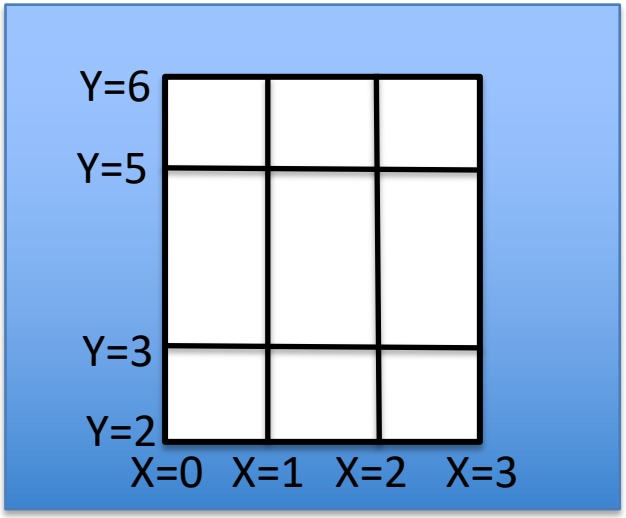
Decisions

- Enumerate all vertices or just locations in X and Y
- Row major versus column major
- (T1, P1, T2, P2, T3, P3, ..., TN, PN) vs.
  (T1, T2, ..., TN) & (P1, P2, ..., PN)
- Float vs double vs unsigned char

- Some choices are better, some are worse, some are neutral
- Whatever choices we make establish our convention
- The following slides show the conventions we will use for this course

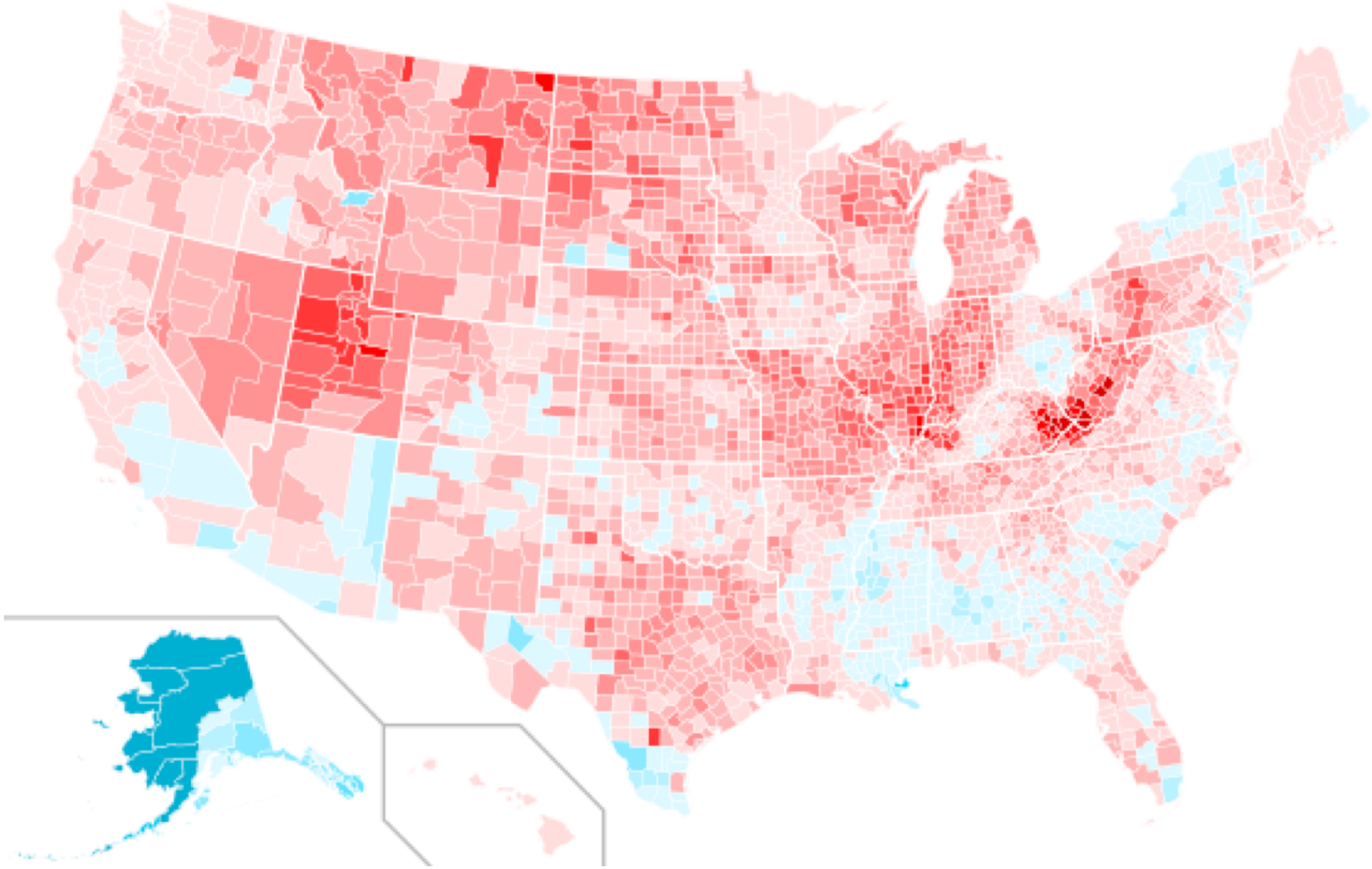
4 Types of Indices

- 4 Types of Indices
  - Point Indices
  - Logical Point Indices
  - Cell Indices
  - Logical Cell Indices



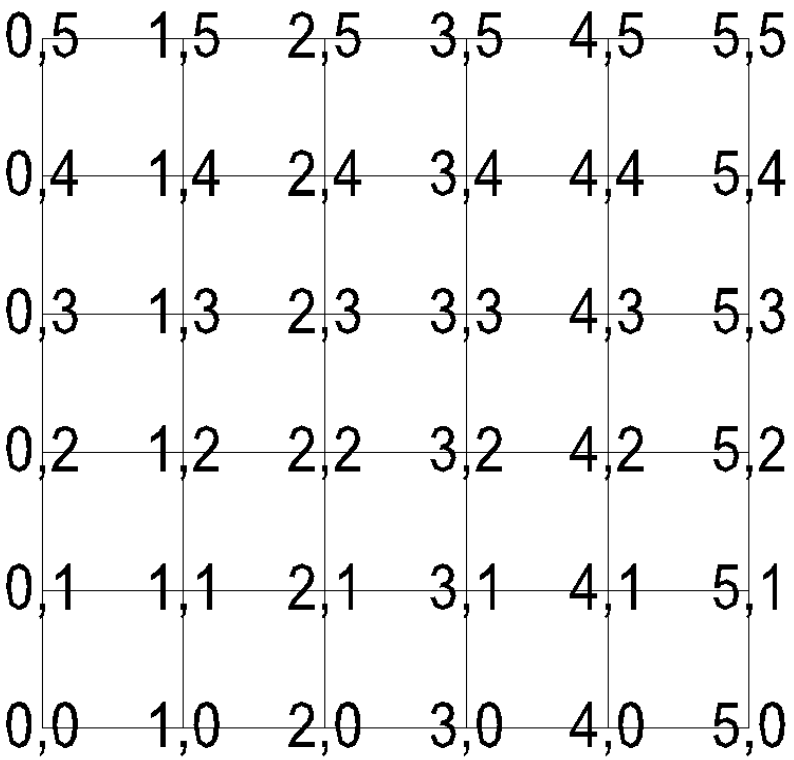
- Point data vs cell data:
  - Point data: fields stored on the “points” of the mesh
  - Cell data: fields stored on the “cells” of the mesh
- Point data is accessed using point indices, cell data is accessed using cell indices
- Logical: an index useful when considering “neighbors”

Cell-centered data

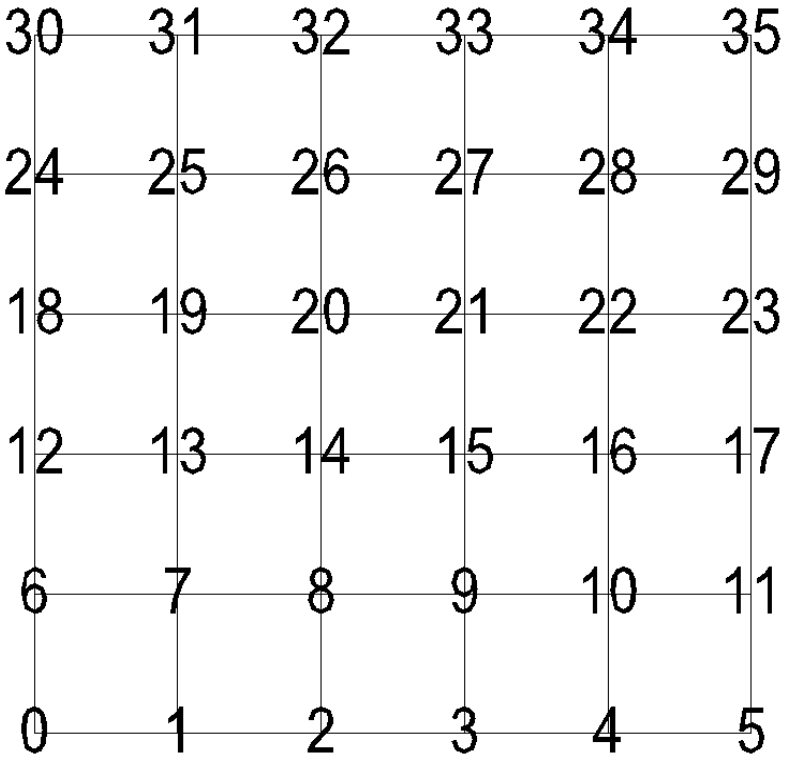


Two schemes for indexing points: "logical point indices" and "point indices"

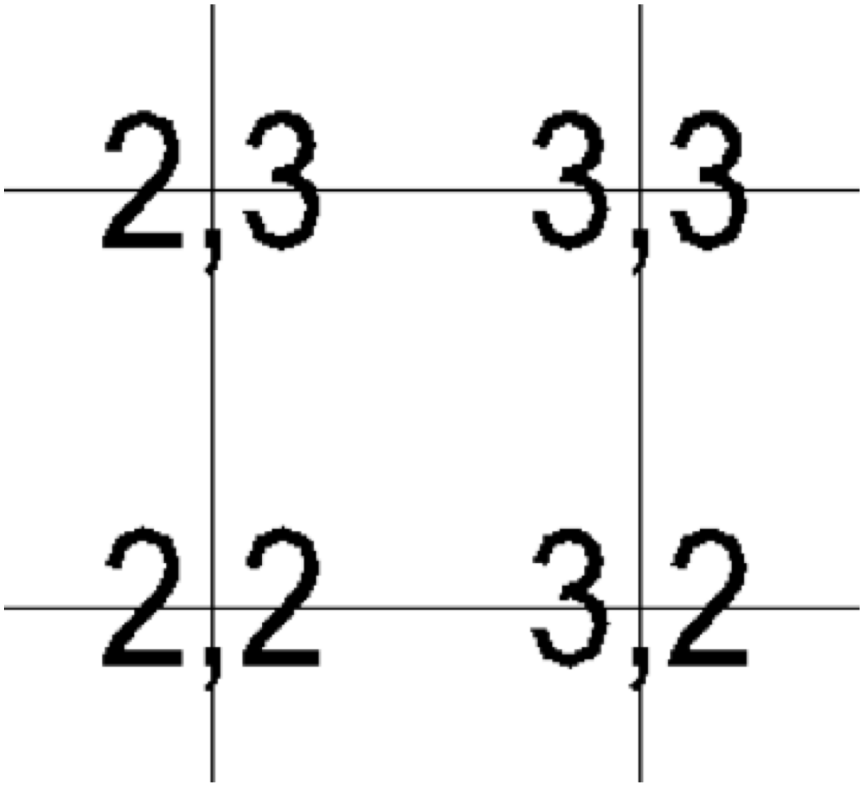
Logical point indices

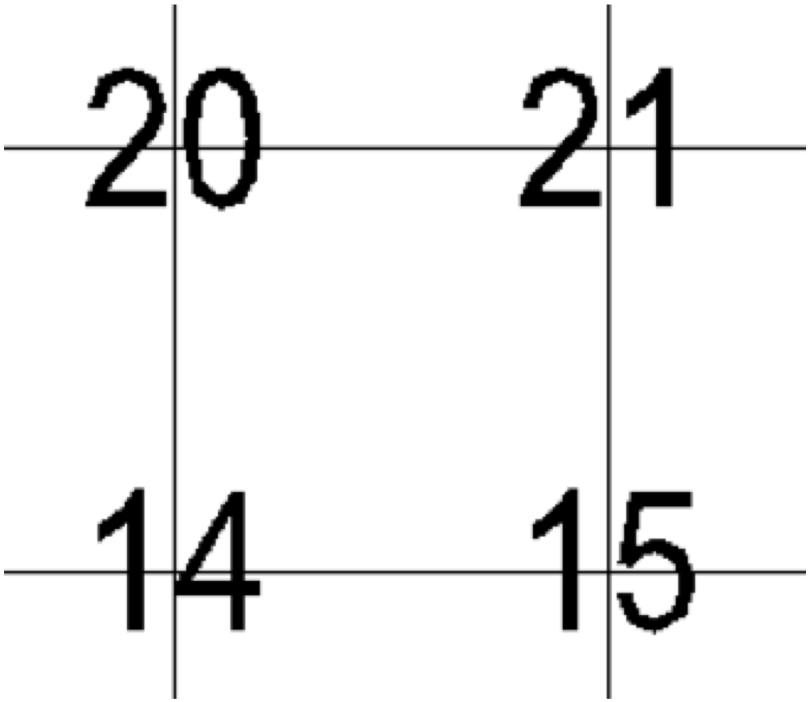


Point indices



Most common usage





Operate on this cell

Access these indices in field array (e.g., temperature array)

How to Index Points

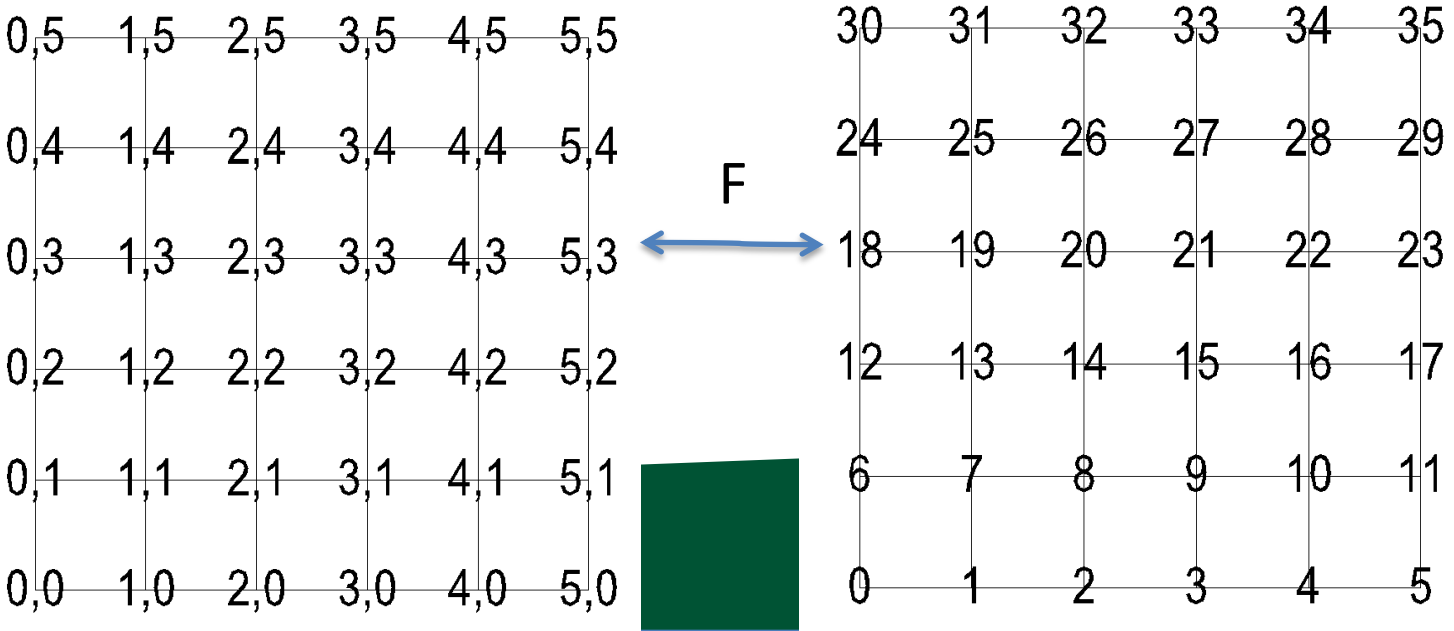
- Our goal: define a bijective function, F, between two sets
  - Set 1 (logical point indices): {(i,j,k): 0<=i<nX, 0<=j<nY, 0<=k<nZ}
  - Set 2 (point indices): {0, 1, …, nPoints-1}

Bijective: for every element in set 1, there is an element in set 2. And vice-versa.

Note: we will focus on 2D rectilinear meshes for a bit.

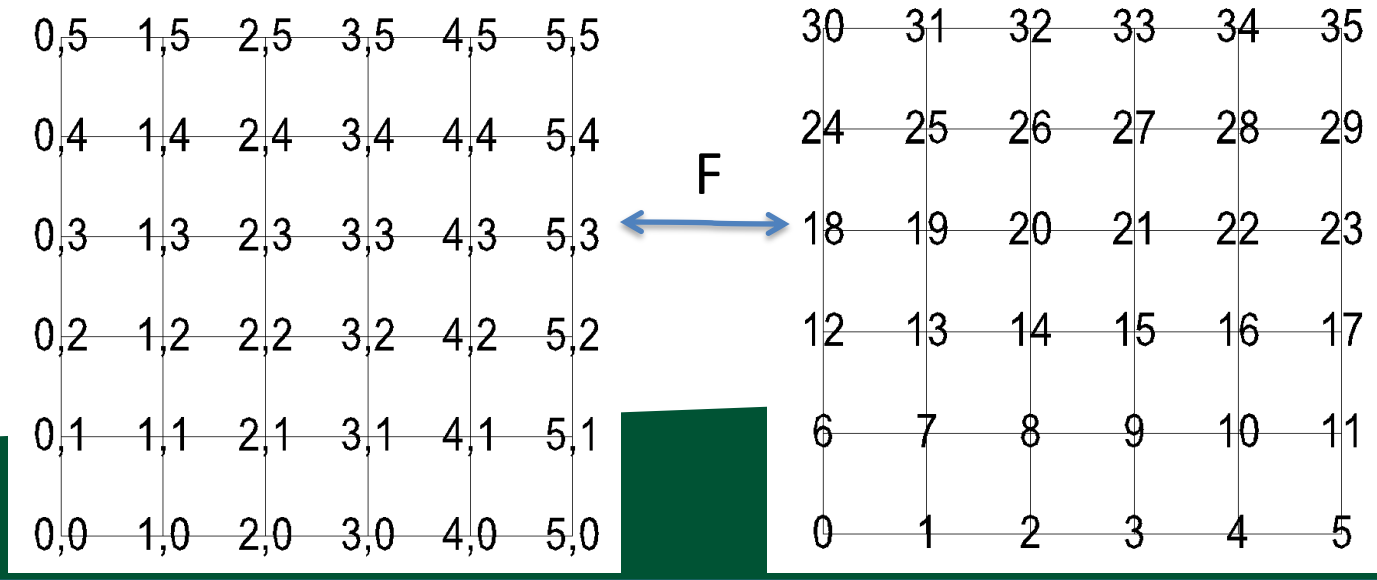
How to Index Points

- Many possible conventions for indexing points and cells
- Most common variants:
  - X-axis varies most quickly
  - X-axis varies most slowly



Bijective function for rectilinear meshes for this course

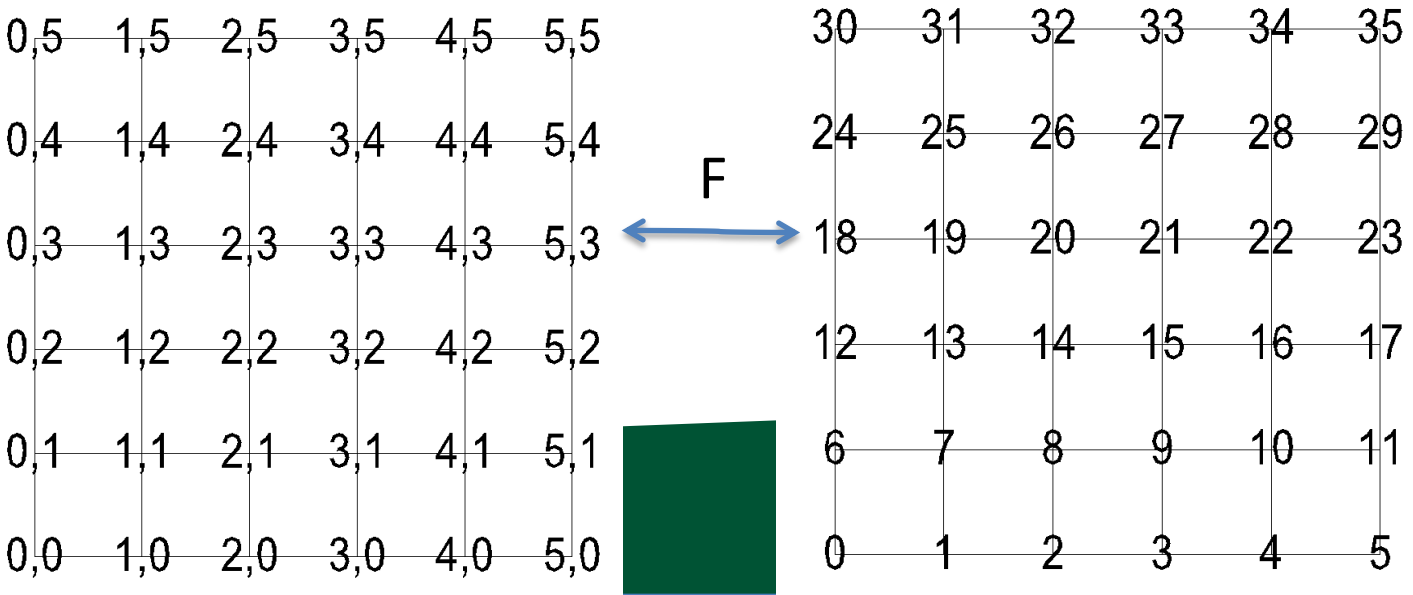
```
int GetPoint(int i, int j,
             int nX, int nY)
{
    return j*nX + i;
}
```



Bijective function for rectilinear meshes for this course

```
int *GetLogicalPointIndex(int point,
                          int nX, int nY)
{
    int  rv[2];
    rv[0] = point % nX;
    rv[1] = (point/nX);
    return rv; // terrible code!!
}
```

```
int *GetLogicalPointIndex(int point,
                          int nX, int nY)
{
    int  rv[2];
    rv[0] = point % nX;
    rv[1] = (point/nX);
    return rv;  // TERRIBLE CODE!
}
```



Quiz Time #2

- A mesh has dimensions 6x8.
- What is the point index for (3,7)? = 45
- What are the logical indices for point 37? = (1,6)

```
int GetPoint(int i, int j,
             int nX, int nY)
{
    return j*nX + i;
}
```

```
int *GetLogicalPointIndex(int point,
                          int nX, int nY)
{
    int  rv[2];
    rv[0] = point % nX;
    rv[1] = (point/nX);
    return rv; // terrible code!!
}
```

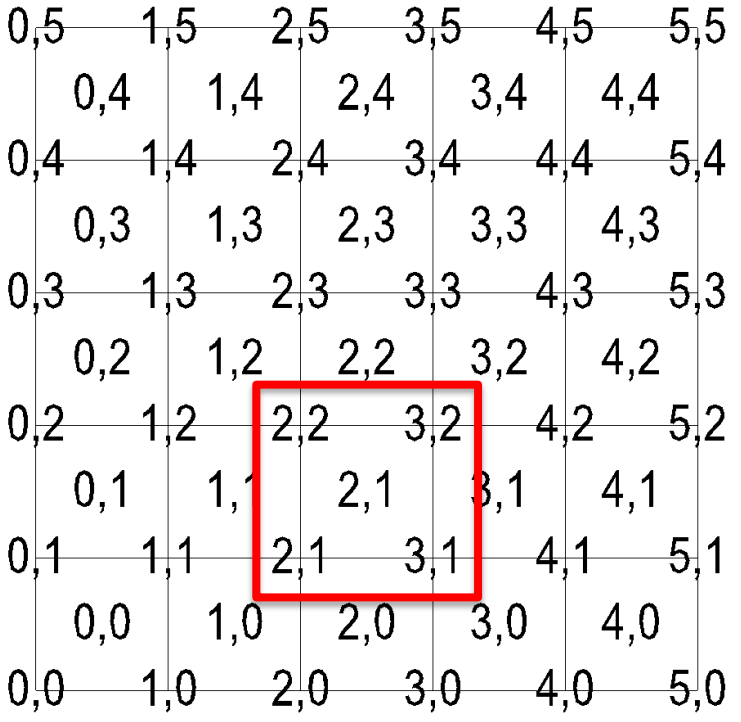
Quiz Time #3

- A vector field is defined on a mesh with dimensions 100x100
- The vector field is defined with double precision data.
- How many bytes to store the vector field?

= 100x100x2x8 = 160,000

Bijective function for rectilinear meshes for this course

```
int GetCell(int i, int j,
            int nX, int nY)
{
  return j*(nX-1) + i;
}
```



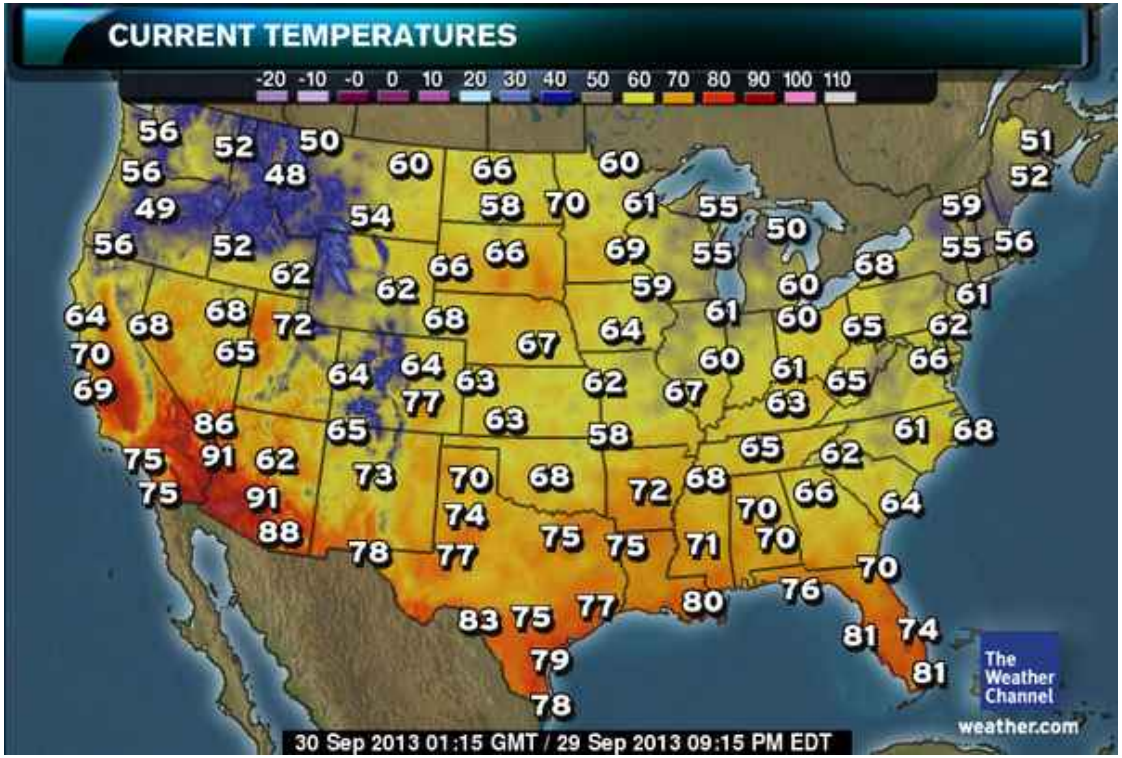
Bijective function for rectilinear meshes for this course

```
int *GetLogicalCellIndex(int cell,
                         int nX, int nY)
{
    int  rv[2];
    rv[0] = cell % (nX-1);
    rv[1] = (cell/(nX-1));
    return rv; // terrible code!!
}
```

Outline

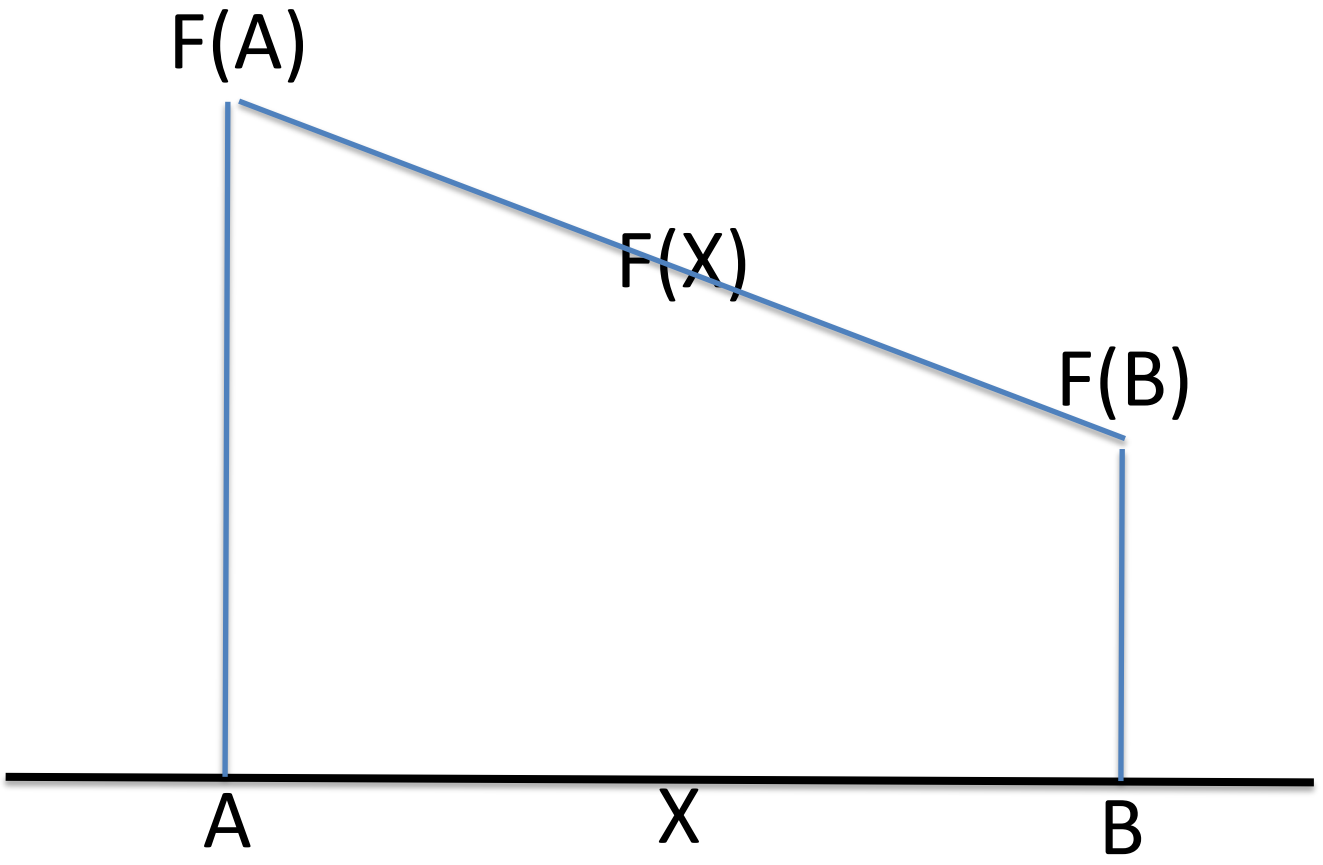
- 6 Slide Review
- The Data We Will Study
  - ~~Overview~~
  - ~~Fields~~
  - Meshes
  - Interpolation

Linear Interpolation for Scalar Field F



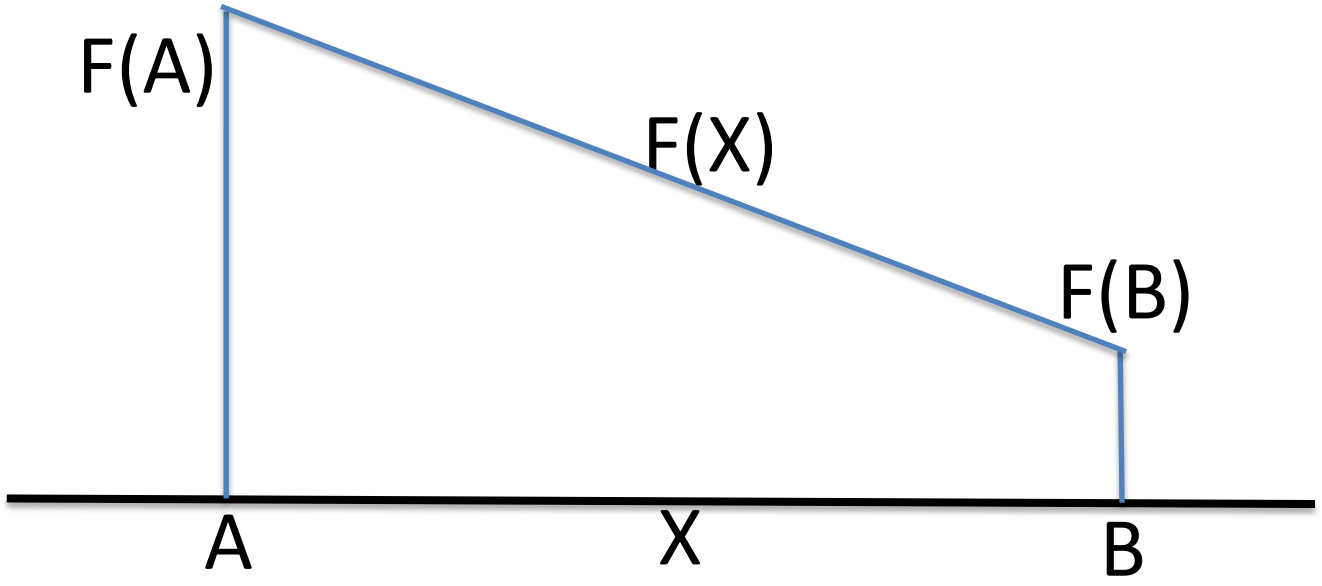
Goal: have data at some points & want to interpolate data to any location

Linear Interpolation for Scalar Field F



Linear Interpolation for Scalar Field F

- General equation to interpolate:
  - F(X) = F(A) + t*(F(B)-F(A))
- t is proportion of X between A and B
  - t = (X-A)/(B-A)

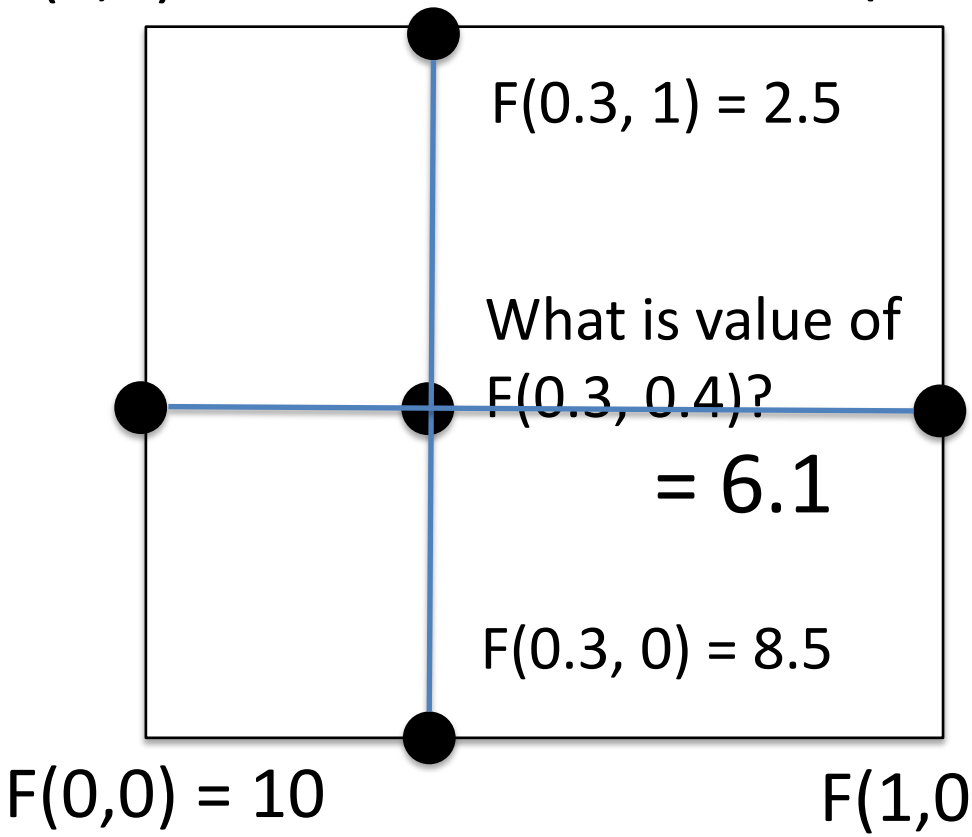


Quiz Time #4

- F(3) = 5, F(6) = 11
- What is F(4)? = 5 + (4-3)/(6-3)*(11-5) = 7

- General equation to interpolate:
  - F(X) = F(A) + t*(F(B)-F(A))
- t is proportion of X between A and B
  - t = (X-A)/(B-A)

Bilinear interpolation for Scalar Field F



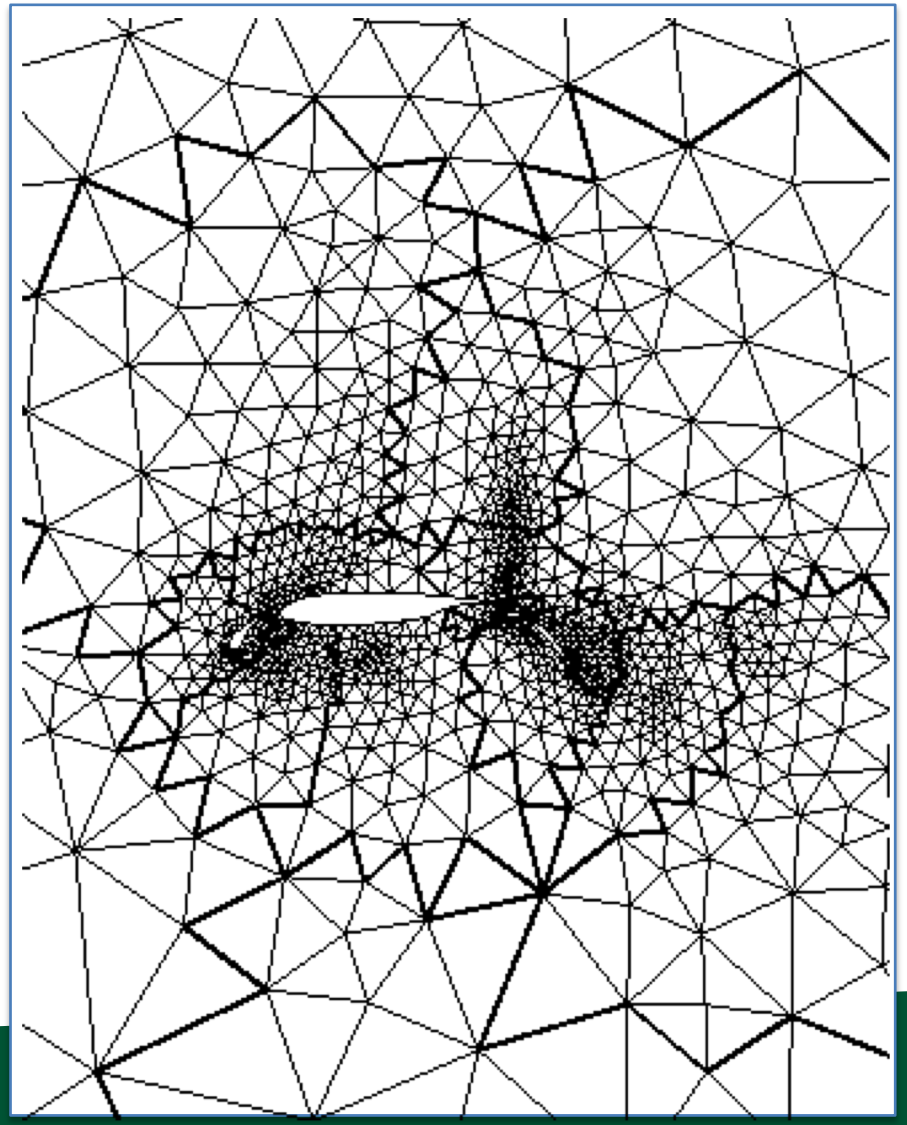
Idea: we know how to interpolate along lines. Let's keep doing that and work our way to the middle.

- General equation to interpolate:

$$F(X) = F(A) + t*(F(B)-F(A))$$

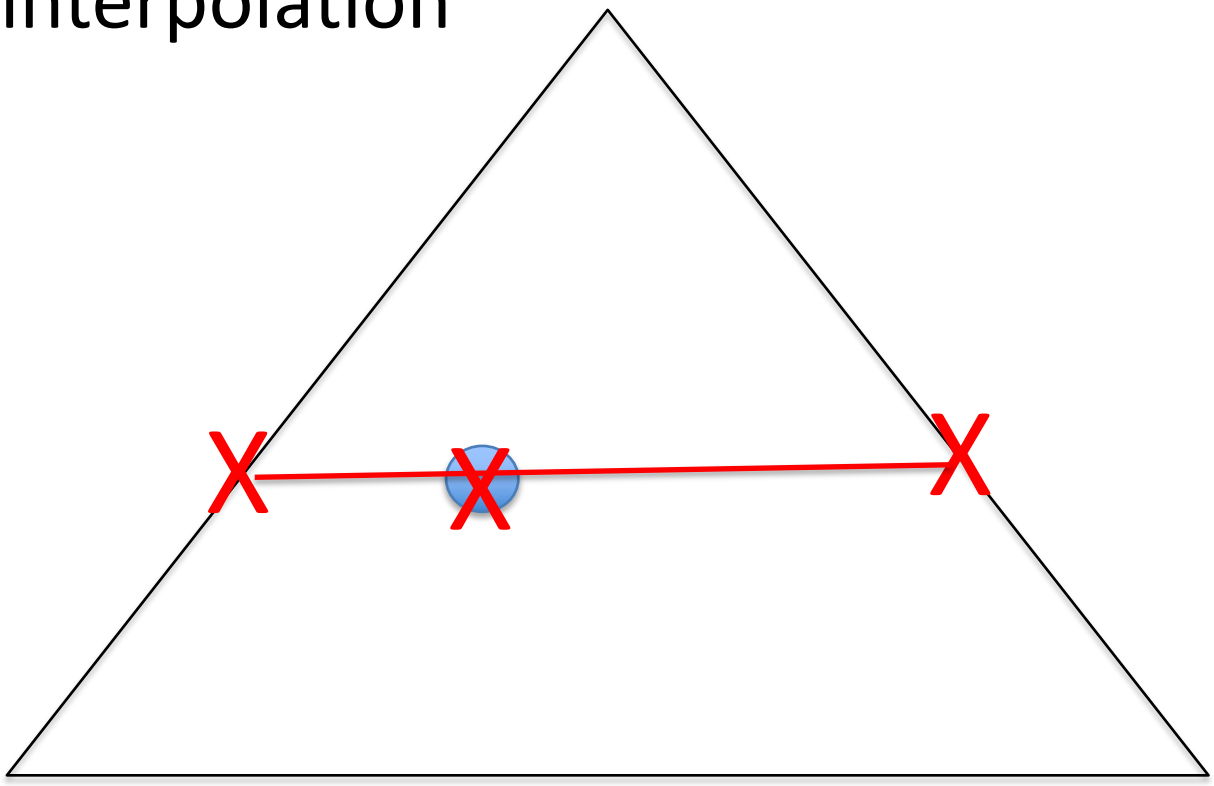
Interpolation for triangle meshes

- Two issues:
  - (1) how to locate triangle that contains point
    - (discuss in 5 slides)
  - (2) how to interpolate to value within triangle
    - (discuss now)



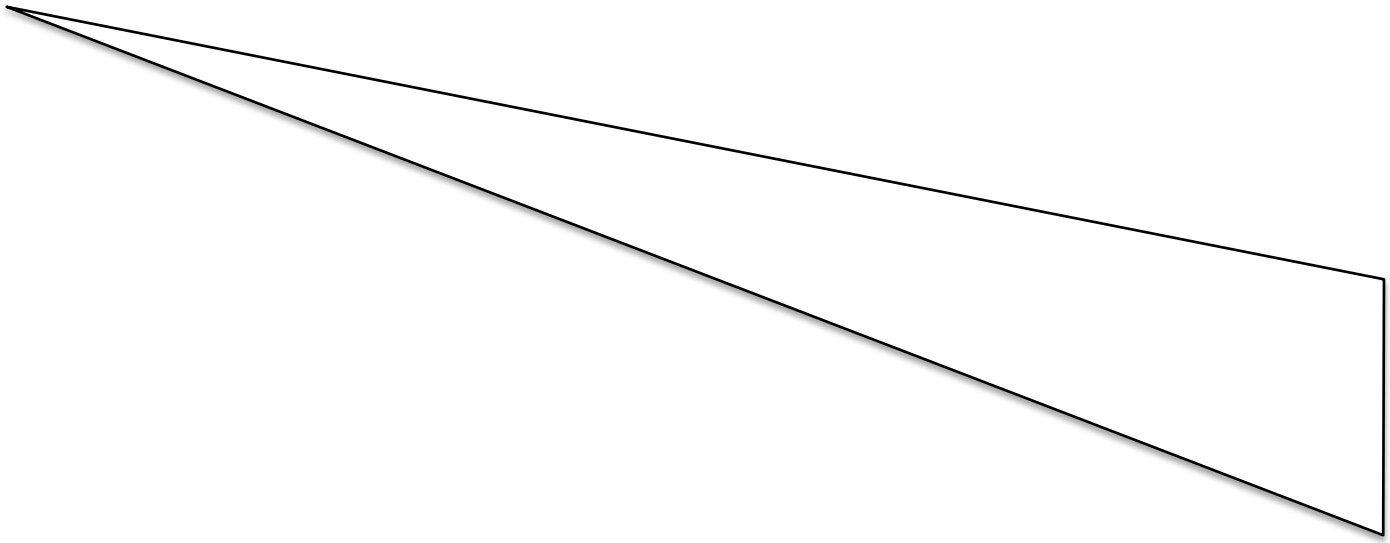
Idea #1

- More bilinear interpolation

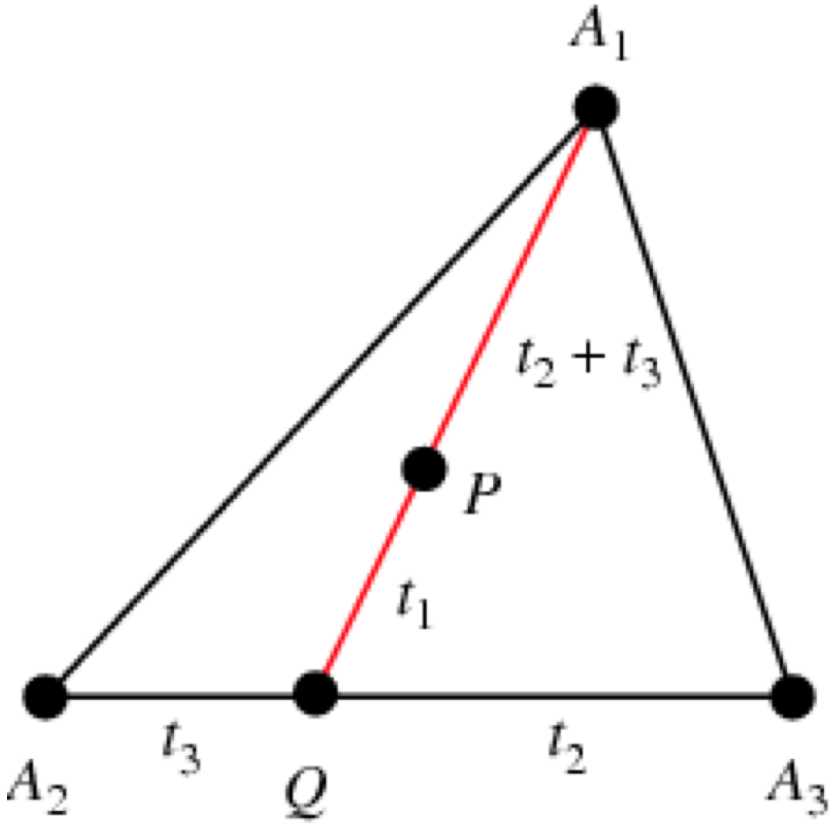


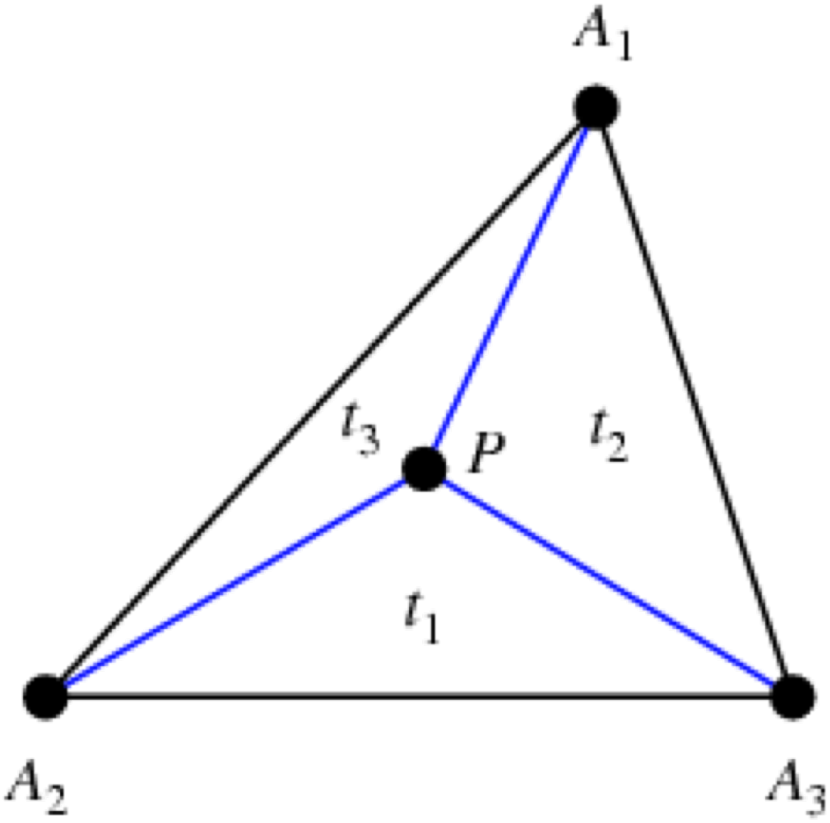
Idea #1 (cont'd)

- Different triangle, similar idea…



Idea #2: Barycentric Coordinates



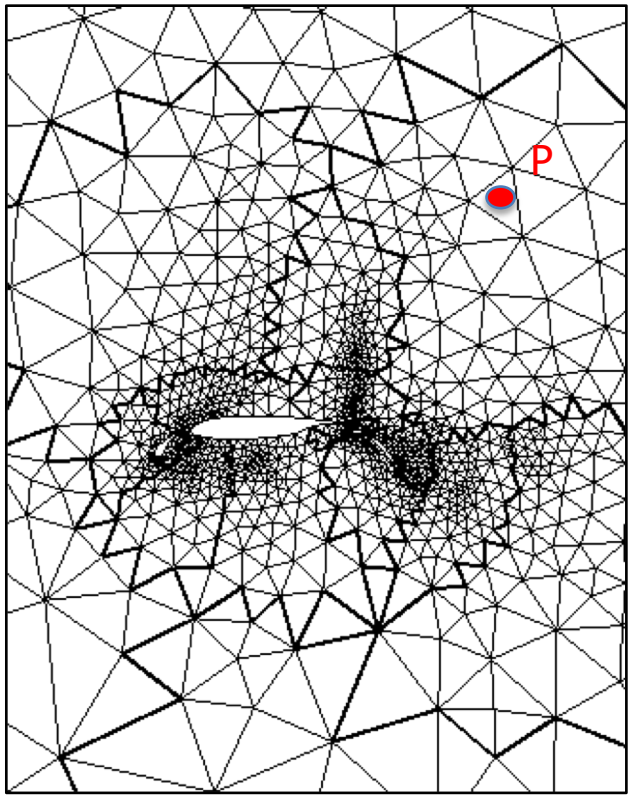


V(P) = V(A1)*t1+V(A2)*t2+V(A3)*t3) / (t1+t2+t3)

http://mathworld.wolfram.com/BarycentricCoordinates.html

Cell location

- Problem definition: you have a physical location (P). You want to identify which cell contains P.



It is easy to identify the cell that contains P with our eyes.
But more work to instruct a computer to do it.
What are your thoughts?

Cell location idea #1 (bad)

- Iterate over every cell
- Check if each cell contains P

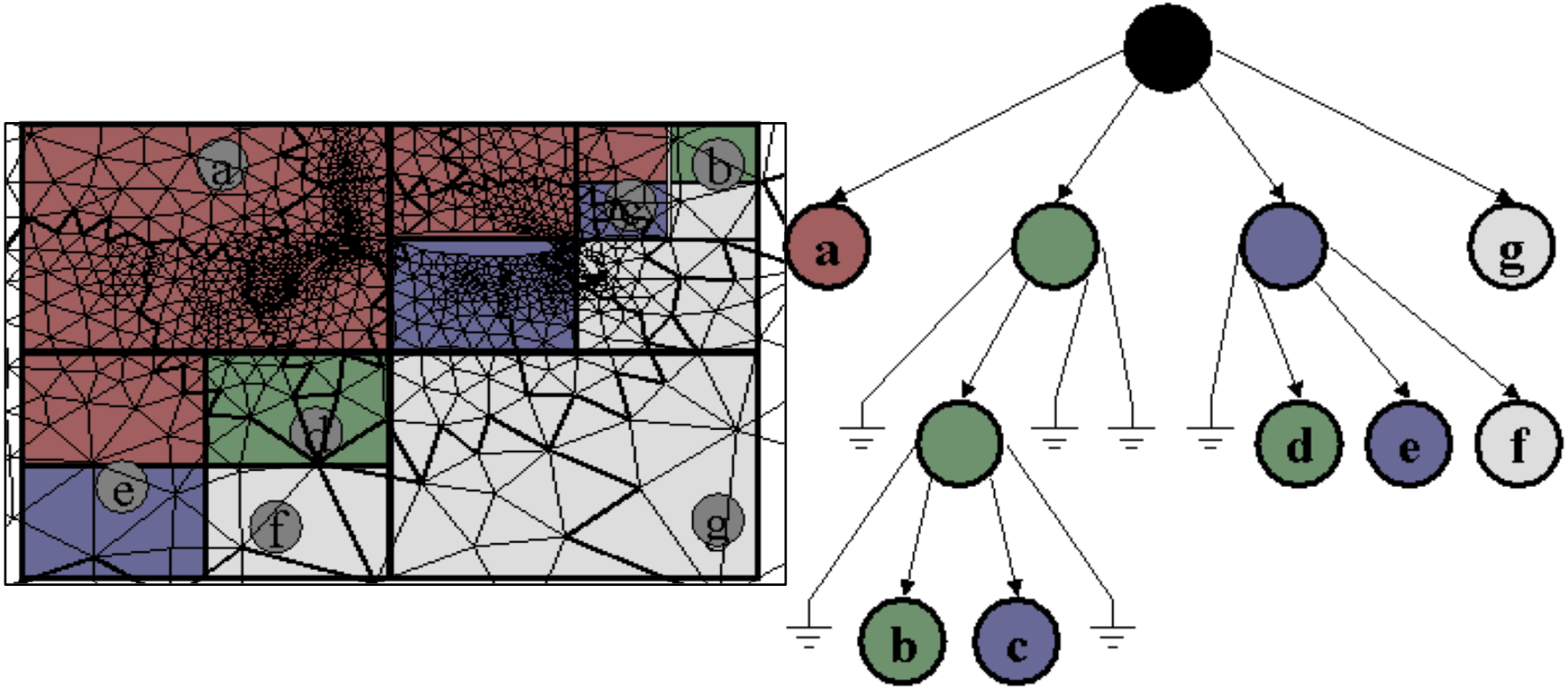
- Setup time: zero
- Search time: O(N), where N is the number of cells
- Search time for M queries: O(M*N)

Cell location idea #2 (good)

- Build “quadtree” data structure
  - (see next slide)
- Takes time to build, but then search is cheap

- Setup time: O(N log N), where N is the number of cells
- Search time: O(log N)
- Search time for M queries:
  - O(M*logN) + O(N log N)

Cell location idea #2 (good)



Credit: courses.cs.washington.edu

Comparing ideas

- Bad idea: O(M*N)
- Good idea:
  - O(N*logN) + O(M*logN)
  - = O((N+M)*logN)
- “Bad idea” is actually better if M very small

Project 2: Field Evaluation

- Assigned today, prompt online
- Due Sun Jan 16 midnight (→ 6am January 17th)
- Worth 5% of your grade
- I provide:
  - Code skeleton online
  - Correct answers provided
- What you upload to Canvas? ... your source code
- Note: Project 3 coming on Thursday