

Postlude

**Done with CS 1110**  
**Where to Next?**

# Announcements

---

## Finishing Up

---

- **Submit a course evaluation**
  - Will get an e-mail for this
  - Part of the “participation grade” (e.g. clicker grade)
- **Final, Dec 10<sup>th</sup> 2:00-4:30pm**
  - Study guide is posted
- **Conflict with Final Exam?**
  - e.g. > 2 finals in 24 hours
  - Submit conflicts **TODAY**

## Review Sessions

---

- **Wednesday 1-4 (Olin 155)**
  - Lists, recursion
  - Open question session
- **Thursday 1-4 (Call Aud)**
  - Invariants, algorithms
  - Open question session
- **Friday 2-5 (Olin 155)**
  - Call frames & diagramming
  - Classes, try-except

# Obvious Next Step: CS 2110

---

- **Programming in Java**

- Basic Java syntax
- Static vs. Dynamic Types
- Adv. Java Topics (e.g. Threads)

- **OO Theory**

- More design patterns
- Interface vs. Implementation

- **Data Structures**

- Binary Trees
- Linked Lists
- Graphs

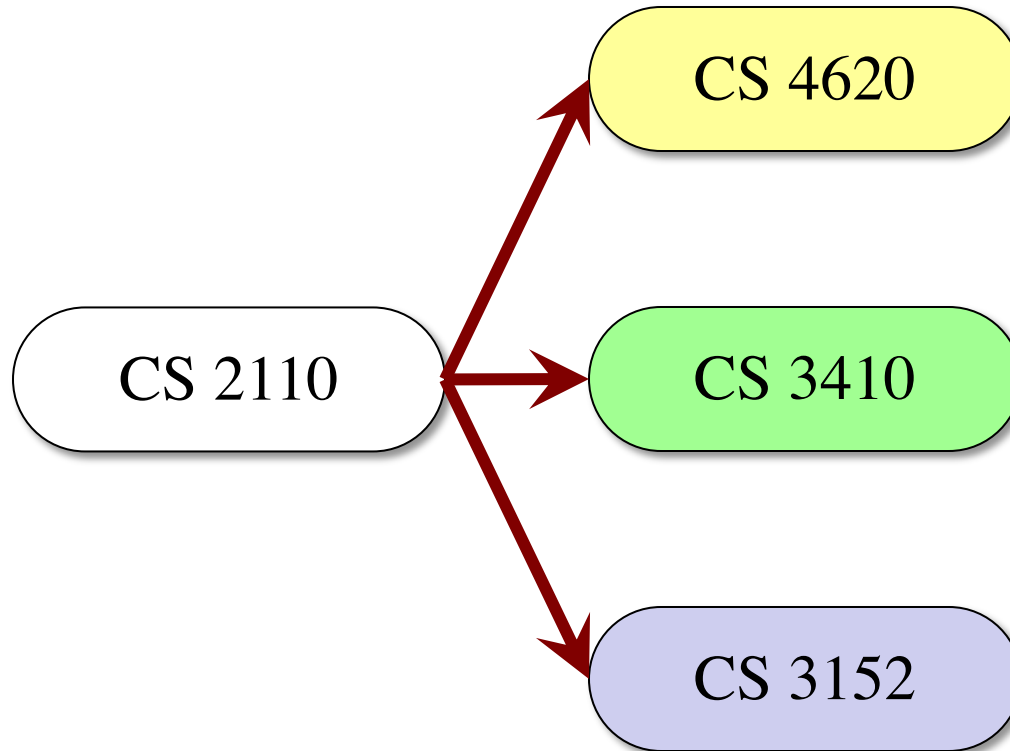
Major CS Topic

Java Specific

Language Independent

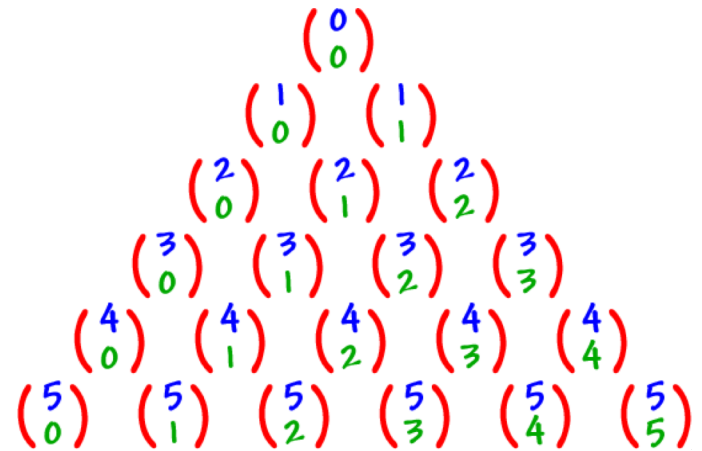
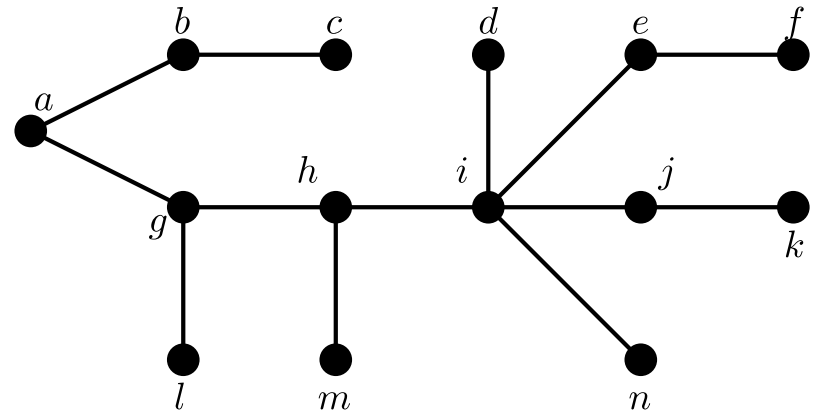
# CS 2110 Immediately Opens your Options

---



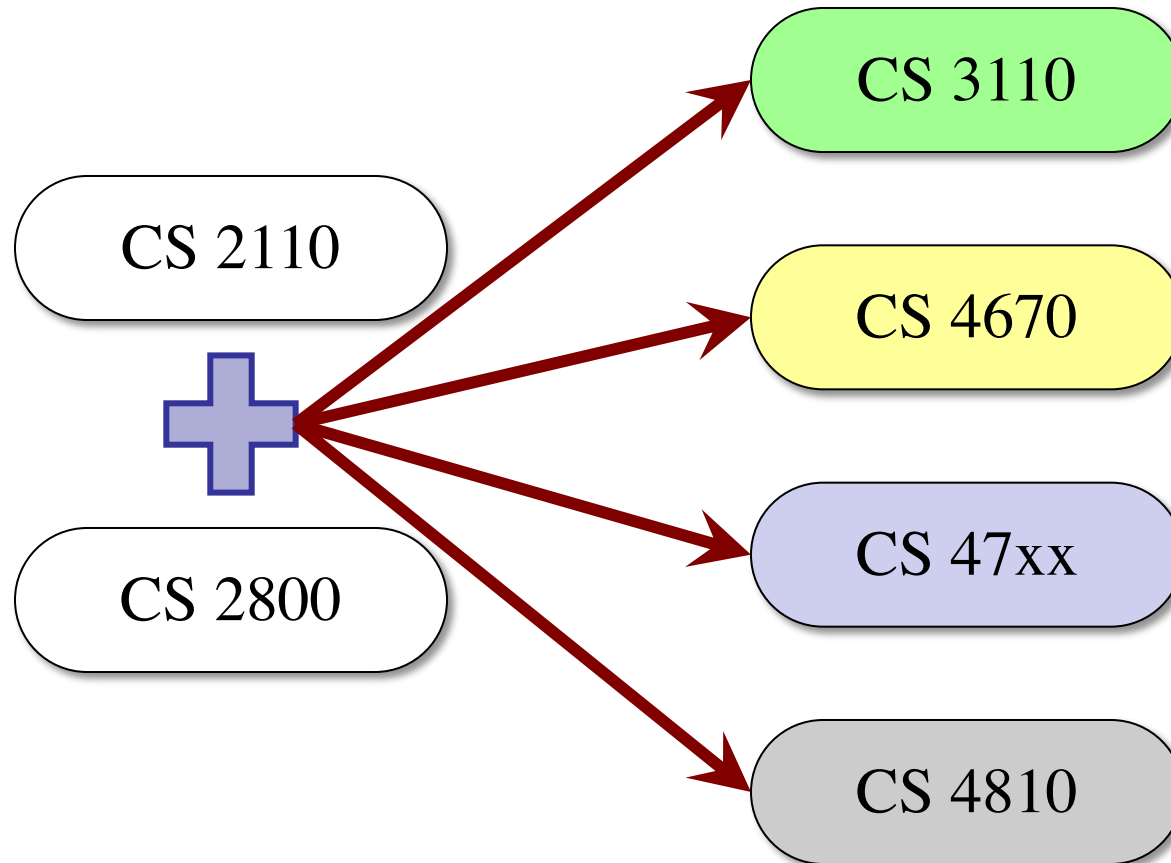
# CS 2800: The Other Important Course

- CS requires a lot of math
  - Analyzing code performance
  - Analyzing data
  - Proving code correctness
- Calculus is “wrong math”
  - Data is rarely “continuous”
  - Limited to specific uses (e.g. spatial data)
- “Grab-bag” course
  - All math needed for CS
  - Includes writing proofs



# CS 2110 + CS 2880 = Even More Options

---



# Higher Level Computer Science Courses

---

- Programming Languages      **x1xx** (e.g. 1110, 2110)
- Scientific Computing      **x2xx** (e.g. 4210)
- Data Management      **x3xx** (e.g. 3300, 4320)
- Systems      **x4xx** (e.g. 3410, 4410)
- Computational Biology      **x5xx** (e.g. 5555)
- Graphics and Vision      **x6xx** (e.g. 4620)
- Artificial Intelligence      **x7xx** (e.g. 4758, 4700)
- Theory      **x8xx** (e.g. 4810, 4820)
- Research      **x9xx** (e.g. 4999)

# Higher Level Computer Science Courses

---

- Programming Languages      **x1xx** (e.g. 1110, 2110)
- Scientific Computing      **x2xx** (e.g. 4210)
- Data Management      **x3xx** (e.g. 4320)
- Systems      **x4xx** (e.g. 4410)
- Computer Architecture      **x5xx** (e.g. 4510)
- Graphics      **x6xx** (e.g. 4620)
- Artificial Intelligence      **x7xx** (e.g. 4758, 4700)
- Theory      **x8xx** (e.g. 4810, 4820)
- Research      **x9xx** (e.g. 4999)

Separation not perfect;  
there is a lot of overlap



# Programming Languages

---

- **Adv. Language Topics**

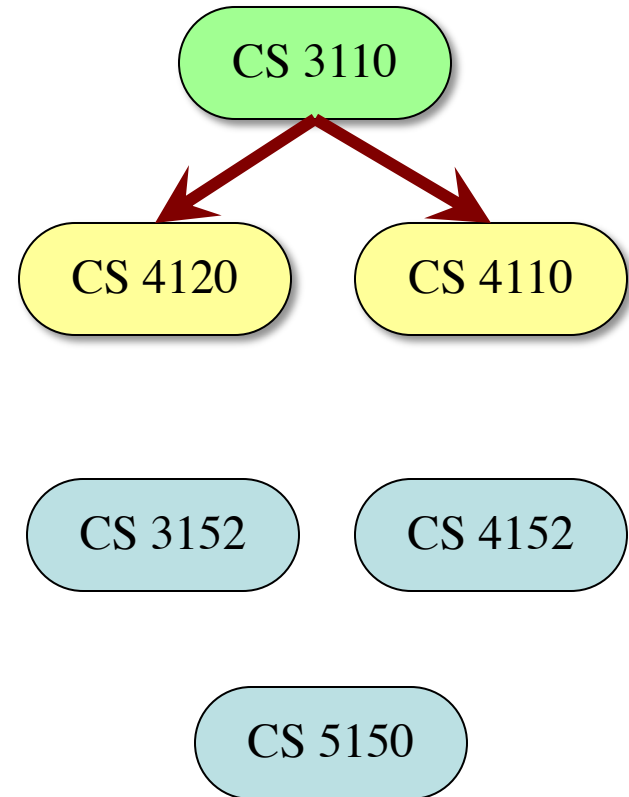
- Functional languages
- Streaming languages
- Parallel programming

- **Language Theory**

- New languages/compilers
- Software verification

- **Software Engineering**

- Design patterns
- Architecture principles



# Scientific Computing

- **Calculus + Computing**

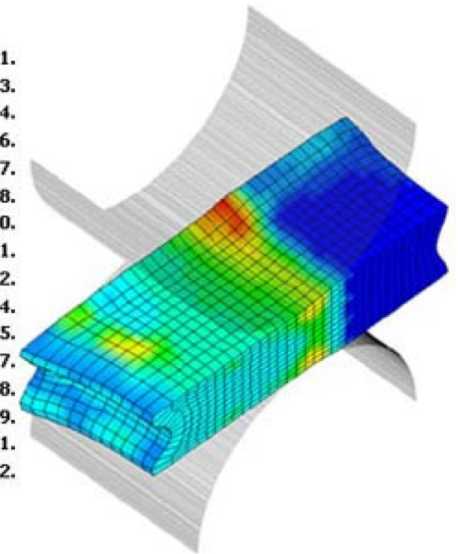
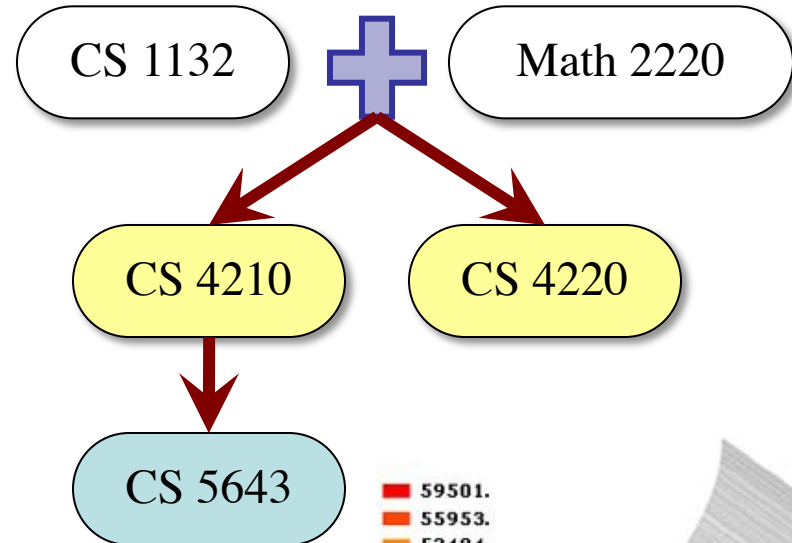
- Problems from other science domains
- Process with computer

- **Applications**

- Complex simulations
- Physics (games!)

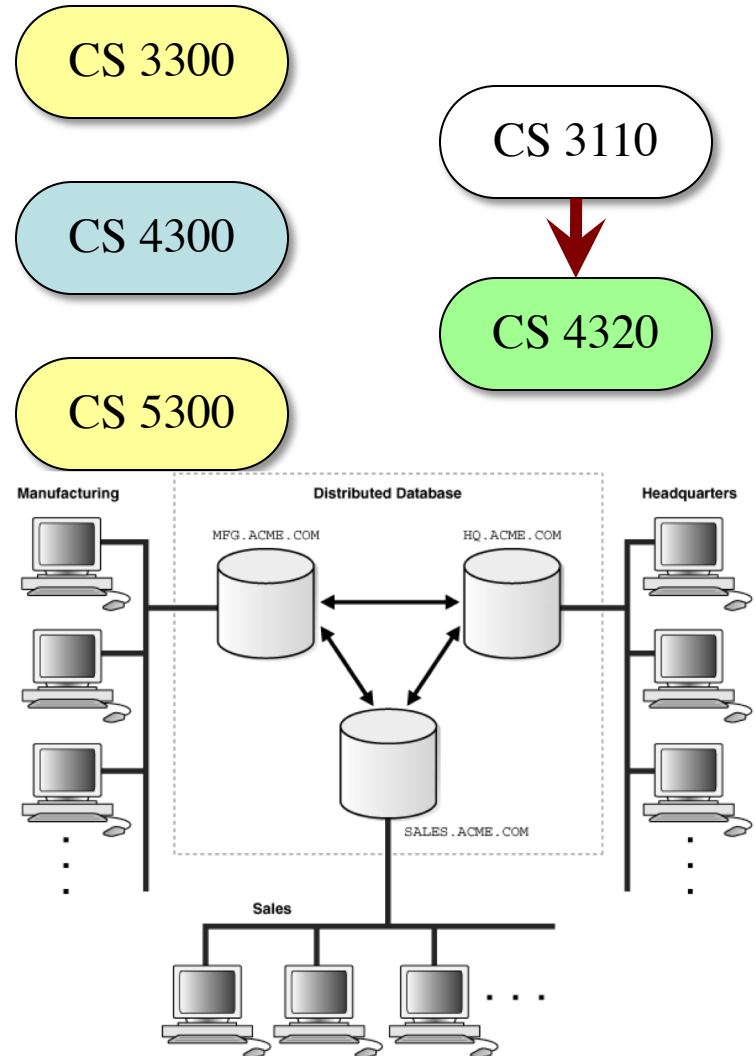
- **Challenge: Performance**

- Programs can run for days!
- How do we make faster?



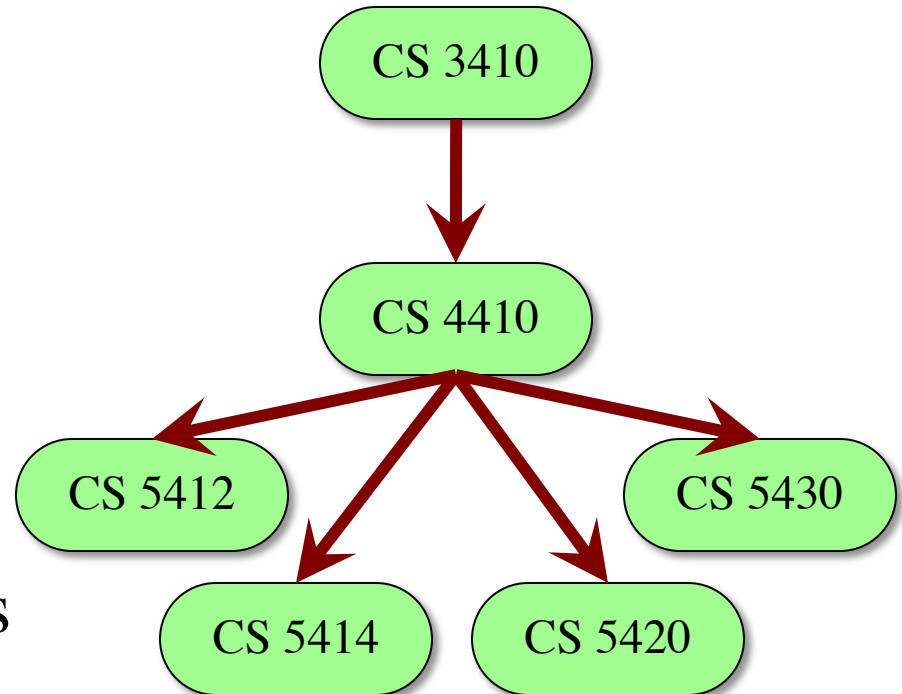
# Data Management

- **Modern Web Apps**
  - Storing user/session data
  - Coordinating users
- **Databases**
  - Query languages
  - Database optimization
  - Organizing your data
- **Information Retrieval**
  - Searching
  - Data analysis



# Systems

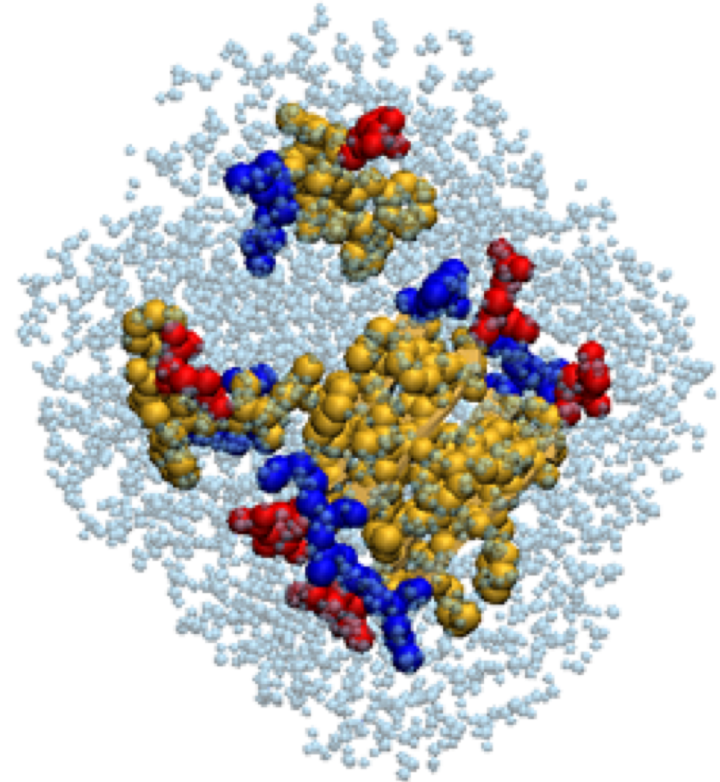
- **Building BIG software**
  - Operating systems
  - Distributed applications (e.g. online, networked)
  - Cloud computing
- Also **System Security**
  - Though that is spread about
- Senior/masters level classes
  - Bulk of the 5xxx courses
  - But great project courses!



# Computation Biology

---

- No undergrad classes
  - Too much to learn
  - Masters/PhD level
- Undergrad options
  - **BTRY 4840:**  
Comp. Genomics
  - BSCB department
- Hoping to improve...



# Graphics and Vision

- **Not** modeling/art!
- **Rendering & Animation**
  - Illumination/reflection
  - Cloth/hair simulation
  - Water and fluids
- **Processing Images**
  - Recognizing shapes
  - Assembling 3D models from 2D pictures
  - Smart cameras

CS 4620

CS 5625

CS 5643

CS 4670



# Artificial Intelligence

- **Not** sentient computers
- **Machine learning**
  - Discovering patterns
  - Making predictions
- **Natural Language Proc.**
  - Automatic translation
  - Searching text/books
  - Voice-control interfaces
- **Robotics**
  - Autonomous control

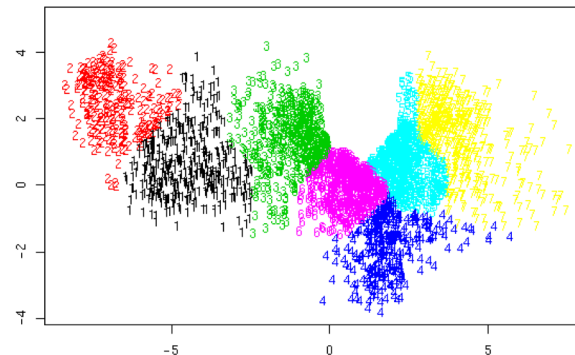
CS 4700

CS 4750

CS 4740

CS 4780

CS 4758



# Theory

- **Analysis of Algorithms**

- What is *possible*?
- What is *feasible*?

CS 4810

CS 4830

CS 4860

CS 4820

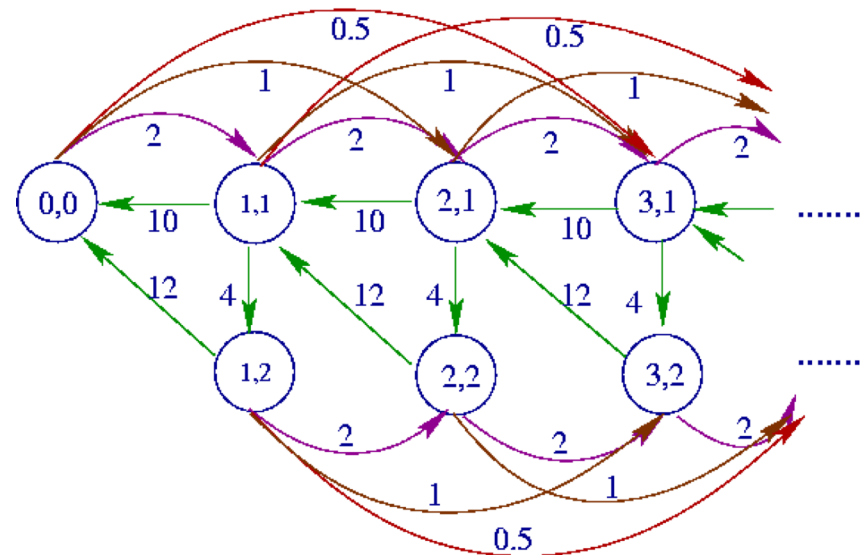
- **Analysis of Structures**

- Social network theory
- Complex data structures

- **Cryptography**

- Theory side of security

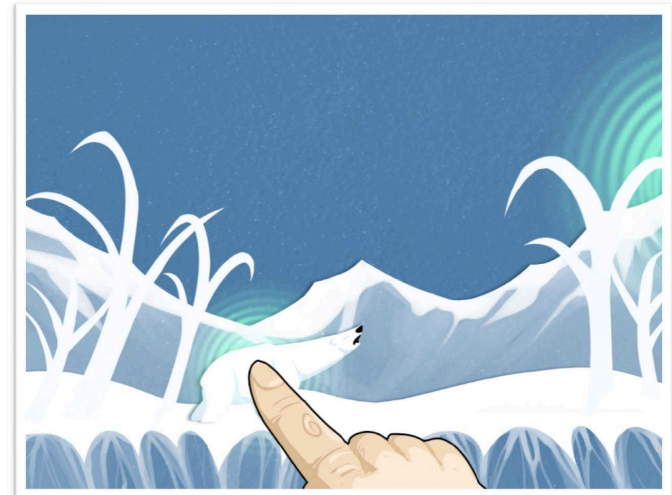
- Perhaps the most famous group in the department





# What About Games?

- CS 3152, Spring only
  - Prereq: CS 2110
  - But CS 3110 a big help
- Build game from scratch
  - Want it to be innovative
  - You own the IP
- Interdisciplinary teams
  - 5 to 6 people on a team
  - With artists/designers
- **Final:** public showcase



# What About Games?

- CS 3152, Software Engineering
  - Prereq: CS 2110
  - But CS 3110 a big help
- Build game from scratch
  - Want it to be innovative
  - You own the IP
- Interdisciplinary teams
  - 5 to 6 people on a team
  - With artists/designers
- **Final:** public showcase



# Games and the Designer Track

- Coding not your thing?
- INFO 3152 (co-meets)
  - Artists/designer track
  - No formal training needed
  - Submit me a portfolio
- Recommend: INFO 2450
  - Start of the HCI sequence
  - How design effects the user experience
  - Fall course; no prereqs



**Good Bye!**