Life after CS 1110

CS 1110
Introduction to Computing Using Python

[E. Andersen, A. Bracy, D. Fan, D. Gries, L. Lee, S. Marschner, C. Van Loan, W. White]
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

- **Java Specific**
- **Language Independent**
- **Major CS Topic**
CS 2110 Immediately Opens your Options

CS 2110

CS 4620 - Introduction to Computer Graphics
CS 3410 - Computer System Organization and Programming
CS 3152 - Introduction to Computer Game Development
CS 4740 - Natural Language Processing
CS 2800: The Other Important Course

- CS requires a lot of math
  - Analyzing code performance
  - Analyzing data
  - Proving code correctness
- Calculus not the only math
  - Data often not “continuous”
  - Limited to specific uses (e.g. spatial data)
- “Grab-bag” course
  - All math needed for CS
  - Includes writing proofs
CS 2110 + CS 2800 = Even More Options

- CS 3110: Data Structures and Functional Programming
- CS 4670: Introduction to Computer Vision
- CS 47xx: Artificial Intelligence, Robotics, Machine Learning (some non-cs pre-reqs)
- CS 4810: Introduction to Theory of Computing
Computer Science Course Numbers

- Programming Languages  x1xx  (e.g. 1110, 2110)
- Scientific Computing  x2xx  (e.g. 3220, 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)
## Computer Science Course Numbers

<table>
<thead>
<tr>
<th>Level</th>
<th>Area</th>
<th>Example Courses</th>
</tr>
</thead>
<tbody>
<tr>
<td>x1xx</td>
<td>Programming Languages</td>
<td>e.g. 1110, 2110</td>
</tr>
<tr>
<td>x2xx</td>
<td>Scientific Computing</td>
<td>e.g. 3220, 4210</td>
</tr>
<tr>
<td>x3xx</td>
<td>Data Management</td>
<td>e.g. 3300, 4320</td>
</tr>
<tr>
<td></td>
<td>Systems</td>
<td>e.g. 3310, 4310</td>
</tr>
<tr>
<td>x4xx</td>
<td>Comput. Engineering</td>
<td>e.g. 3340, 4353</td>
</tr>
<tr>
<td>x6xx</td>
<td>Graphics</td>
<td>e.g. 4620</td>
</tr>
<tr>
<td>x7xx</td>
<td>Artificial Intelligence</td>
<td>e.g. 4758, 4700</td>
</tr>
<tr>
<td>x8xx</td>
<td>Theory</td>
<td>e.g. 4810, 4820</td>
</tr>
<tr>
<td>x9xx</td>
<td>Research</td>
<td>e.g. 4999</td>
</tr>
</tbody>
</table>
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!
Computational Health/Biology

• No undergrad classes
  - Too much to learn
  - Masters/PhD level

• Undergrad options
  - **BTRY 4840:**
    - Comp. Genomics
  - BSCB department

• Stay tuned for more…
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

Courses:
- 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
Theory

• Analysis of Algorithms
  ▪ What is possible?
  ▪ What is feasible?

• Analysis of Structures
  ▪ Social network theory
  ▪ Complex data structures

• Cryptography
  ▪ Theory side of security
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
Games and the Designer Track

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

**1110**: Introduction to Computing Using Python
**1112**: Introduction to Computing Using MATLAB
**1132**: Short Course in MATLAB
**1133**: Short Course in Python
**1380**: Data Science for All
**2024**: C++ Programming

**2110**: Object-Oriented Programming and Data Structures
**2112**: Object-Oriented Design and Data Structures - Honors
**2770**: Excursions in Computational Sustainability

**3110**: Data Structures and Functional Programming
**3152**: Introduction to Computer Game Architecture
**3220**: Introduction to Scientific Computation
**3410**: Computer System Organization and Programming
**3420**: Embedded Systems (prereq: ENGRD 2300, not shown)
**4110**: Programming Languages and Logics
**4120**: Introduction to Compilers
**4152**: Advanced Topics in Computer Game Architecture
**4154**: Analytics-driven Game Design
**4160**: Formal Verification
**4220**: Numerical Analysis: Linear and Nonlinear Problems
**4320**: Introduction to Database Systems
**4410**: Operating Systems
**4450**: Introduction to Computer Networks
**4620**: Introduction to Computer Graphics
**4670**: Introduction to Computer Vision
**4700**: Foundations of Artificial Intelligence
**4740**: Natural Language Processing
**4750**: Foundations of Robotics
**4780**: Machine Learning for Intelligent Systems
**4786**: Machine Learning for Data Science
**4787**: Principles of Large-Scale Machine Learning
**4810**: Introduction to Theory of Computing
**4820**: Introduction to Analysis of Algorithms
**4850**: Mathematical Foundations for the Information Age
**4860**: Applied Logic
Computer Science not your ☕️?  

Try one of our neighbors!  
• Information Science  
• Statistics  
• Operations Research & Information Engineering  
• Electrical and Computer Engineering  
  ▪ ECE 2400 is a good next step
It's been a challenging semester given the state of the world and everyone’s individual situation.

Thank you for persevering!!!!

Hope you’ve found some parts of CS1110 interesting and will find some parts useful in the future!