Introduction to Programming in Python
Course Mechanics
Website

https://www.swamiiyer.net/cs110

What's on the Site?

• Announcements (landing page)
• Course Info
• Calendar
• Lecture Material
• Assignments
• Resources
Goal

Attain proficiency in the design and implementation of Python programs of significant size and complexity
Attain proficiency in the design and implementation of Python programs of significant size and complexity
Prerequisites

Math 140 (Calculus I) credits or placement or Math 130 (Precalculus) with a B or higher in the previous semester or Permission of the instructor
Prerequisites

Math 140 (Calculus I) credits or placement or

Math 130 (Precalculus) with a B or higher in the previous semester or

Permission of the instructor
Name: Swami Iyer (Senior Lecturer, Computer Science Department)

Academic Interests: Evolutionary dynamics on complex networks, coding, pedagogy

Contact Information:
• Office: M-3-201-14
• Email: siyer@cs.umb.edu (start subject line with [CS110])

Office Hours:
• Tue Thu 9:30 AM – 10:30 AM and 2:30 PM – 3:30 PM (in-person)
• Wed 10:00 AM – 12:00 PM (remote)
Name: Swami Iyer (Senior Lecturer, Computer Science Department)

Academic Interests: Evolutionary dynamics on complex networks, coding, pedagogy

Contact Information:
- Office: M-3-201-14
- Email: siyer@cs.umb.edu (start subject line with [CS110])

Office Hours:
- Tue Thu 9:30 AM – 10:30 AM and 2:30 PM – 3:30 PM (in-person)
- Wed 10:00 AM – 12:00 PM (remote)
<table>
<thead>
<tr>
<th>Session</th>
<th>Class When</th>
<th>Where</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 – 8</td>
<td>Tue Thu 11:00 AM – 12:15 PM</td>
<td>W-1-0088 (Snowden Auditorium)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Discussion</th>
<th>Section When</th>
<th>Where</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tue 9:30 AM – 10:45 AM</td>
<td>M-1-0209</td>
</tr>
<tr>
<td>2</td>
<td>Thu 9:30 AM – 10:45 AM</td>
<td>M-2-0206</td>
</tr>
<tr>
<td>3</td>
<td>Tue 12:30 PM – 1:45 PM</td>
<td>M-2-0211</td>
</tr>
<tr>
<td>4</td>
<td>Thu 12:30 PM – 1:45 PM</td>
<td>W-2-0198</td>
</tr>
<tr>
<td>5</td>
<td>Tue 9:30 AM – 10:45 AM</td>
<td>M-2-0206</td>
</tr>
<tr>
<td>6</td>
<td>Thu 9:30 AM – 10:45 AM</td>
<td>W-1-0061</td>
</tr>
<tr>
<td>7</td>
<td>Tue 12:30 PM – 1:45 PM</td>
<td>W-1-0055</td>
</tr>
<tr>
<td>8</td>
<td>Thu 12:30 PM – 1:45 PM</td>
<td>M-3-0732</td>
</tr>
</tbody>
</table>

**Supplemental Instruction (SI):** details to be determined

**Tutoring:** available through Subject Tutoring Program
## Sessions

### Class:

<table>
<thead>
<tr>
<th>Section</th>
<th>When</th>
<th>Where</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 – 8</td>
<td>Tue Thu 11:00 AM – 12:15 PM</td>
<td>W-1-0088 (Snowden Auditorium)</td>
</tr>
</tbody>
</table>

### Discussion:

<table>
<thead>
<tr>
<th>Section</th>
<th>When</th>
<th>Where</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tue 9:30 AM – 10:45 AM</td>
<td>M-1-0209</td>
</tr>
<tr>
<td>2</td>
<td>Thu 9:30 AM – 10:45 AM</td>
<td>M-2-0206</td>
</tr>
<tr>
<td>3</td>
<td>Tue 12:30 PM – 1:45 PM</td>
<td>M-2-0211</td>
</tr>
<tr>
<td>4</td>
<td>Thu 12:30 PM – 1:45 PM</td>
<td>W-2-0198</td>
</tr>
<tr>
<td>5</td>
<td>Tue 9:30 AM – 10:45 AM</td>
<td>M-2-0206</td>
</tr>
<tr>
<td>6</td>
<td>Thu 9:30 AM – 10:45 AM</td>
<td>W-1-0061</td>
</tr>
<tr>
<td>7</td>
<td>Tue 12:30 PM – 1:45 PM</td>
<td>W-1-0055</td>
</tr>
<tr>
<td>8</td>
<td>Thu 12:30 PM – 1:45 PM</td>
<td>M-3-0732</td>
</tr>
</tbody>
</table>

Supplemental Instruction (SI): details to be determined

Tutoring: available through Subject Tutoring Program
INTRODUCTION TO
Programming
in Python
An Interdisciplinary Approach
Robert Sedgewick • Kevin Wayne • Robert Dondero
<table>
<thead>
<tr>
<th>Grading Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Assessment % of Final Grade</strong></td>
</tr>
<tr>
<td>Programming Assignments (best 5 out of 6)</td>
</tr>
<tr>
<td>Exams (1 and 2)</td>
</tr>
<tr>
<td>Participation</td>
</tr>
</tbody>
</table>

*Programming assignment: interesting computational problems*

*Exam: programming (Oct 22/24 and Dec 3/5) and written (Oct 31 and Dec 12)*

*Participation: in-class quizzes (lowest 5 scores dropped, 8%) and discussion attendance (2%)*

*Up to 1% extra points for unused late days*

*Up to 2% extra points for attending the SI sessions*

*Up to 0.01% extra points if x% of the class completes the end-of-semester course evaluation*

*If overall score is within 0.5% of a higher grade, it will be elevated to that grade*
Grading Scheme

<table>
<thead>
<tr>
<th>Assessment</th>
<th>% of Final Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming Assignments (best 5 out of 6)</td>
<td>40</td>
</tr>
<tr>
<td>Exams (1 and 2)</td>
<td>50</td>
</tr>
<tr>
<td>Participation</td>
<td>10</td>
</tr>
</tbody>
</table>

Programming assignment: interesting computational problems

Exam: programming (Oct 22/24 and Dec 3/5) and written (Oct 31 and Dec 12)

Participation: in-class quizzes (lowest 5 scores dropped, 8%) and discussion attendance (2%)

Up to 1% extra points for unused late days

Up to 2% extra points for attending the SI sessions

Up to 0.01x% extra points if x% of the class completes the end-of-semester course evaluation

If overall score is within 0.5% of a higher grade, it will be elevated to that grade
Software

iClicker

Piazza

Gradescope

Programming environment

Zoom
Software

- iClicker
- Piazza
- Gradescope
- Programming environment
- Zoom
Policies

Classroom

Piazza

Makeup Exam

Late Days

Regrade Request

Collaboration

Accommodations for students with disabilities

Campus Closure
Immediate Action Items

Sign up for iClicker

Sign up for Piazza

Sign up for Gradescope

Setup the programming environment

Fill out the questionnaire available on Gradescope

Complete the SI poll

Sign up for CS account

Math primer

Command-line primer
Topics Covered

Chapter 1: Building a Computer

Chapter 2: Imperative Programming
• Your First Programs
• Basic Data Types
• Control Flow
• Collection Data Types
• Input and Output

Chapter 3: Procedural Programming
• Defining Functions
• Libraries and Applications
• Recursion
Topics Covered

Chapter 1: Building a Computer

Chapter 2: Imperative Programming
- Your First Programs
- Basic Data Types
- Control Flow
- Collection Data Types
- Input and Output

Chapter 3: Procedural Programming
- Defining Functions
- Libraries and Applications
- Recursion
Topics Covered

Chapter 4: Object-oriented Programming
- Using Data Types
- Defining Data Types
- Design Principles

Chapter 5: Algorithms and Data Structures
- Analysis of Algorithms
- Searching and Sorting
- Basic Data Structures
Topics Covered

Chapter 4: Object-oriented Programming
• Using Data Types
• Defining Data Types
• Design Principles

Chapter 5: Algorithms and Data Structures
• Analysis of Algorithms
• Searching and Sorting
• Basic Data Structures