

Principles of Programming Languages

CMSC 331, Spring 2012

<http://cs.umbc.edu/331/spring12>

Staff

- John Y. Park
 - park@umbc.edu
 - ITE 201A, office hours: TTh 4:00–5:00pm
- TA Petr Babkin
 - t57538@umbc.edu
 - ITE 349, office hours: TBD

Goals

- Cover basic concepts underlying programming languages (syntax, semantics, families, ...)
- Study Scheme as an example of a functional programming language
- Study an language such as C or Python as an example of a modern imperative language
- Study an interpreter for Scheme in both Scheme and Python
- Touch on other languages and concepts along the way

Approach

- Text has good coverage of the basics
- Augmented with readings and videos from Web
- Learn by doing: frequent homework (45%)
 - 7-10 assignments
- Exams: midterm (20%), final (30%)
- Discussions in class and online (5%)
- Programming assignments must work on gl

Expectations

- Do the assignments
 - On your own
- Hand them in on time
 - It's better to hand it in late than not at all
- Ask questions
 - And/or share thoughts
- Don't be afraid to seek help
- Take pride in your work, including your code

Infrastructure

- Website for schedule, notes, etc.
 - <http://cs.umbc.edu/courses/331/spring12/>
- Blackboard for discussion and grades
- All programming assignments must work on gl.umbc.edu
- If you use your own computer you'll eventually want to download & install
 - [Racket](#) (Scheme)
 - [Python](#)

Need Help?

- We are here to help you learn
- Recommended procedure
 - Think
 - Check book, reading, notes
 - Check the online discussion forum
 - Ask Google or Bing
 - Ask the TA (email, office hours)
 - Ask the instructor (email, office hours)

Academic Integrity

All members of the UMBC community are expected to make a commitment to academic honesty in their own actions and with others. Academic misconduct could result in disciplinary action that may include suspension or dismissal. Here are examples of academic misconduct that are not tolerated at UMBC.

- **Cheating:** Knowingly using or attempting to use unauthorized material, information, or study aids in any academic exercise
- **Fabrication:** Intentional and unauthorized falsification or invention of any information or citation in an academic exercise
- **Facilitating Academic Dishonesty:** Intentionally or knowingly helping or attempting to help another commit an act of academic dishonesty
- **Plagiarism:** Knowingly representing the words or ideas of another as one's own in any academic exercise, including works of art and computer-generated information/images

Questions

- These are some questions for us to think about throughout the course
- At best they have subjective answers

Questions

- How important is programming to CS?
- How important is the choice of language to a programming task?
- What's the best PL? How many should I know?
- Why are new PLs constantly being invented? Why should I learn any of them?
- How will evolving computing hardware (cloud computing, quantum computers) change PLs?
- How long does it take to master a PL?
- What PLs should I know to get the best jobs?