

CSCI 6305:

Foundations of Algorithms, Data and Programming Languages

Class Time: M 6:30 – 9:00 pm

Classroom: EIEAB. 1.212

Instructor: Dr. Andres Figueroa

Office: EIEAS 3.247

OH: MW 10:45 – 11:45 am, M 4:30 – 5:30 pm or by
appointment

Email: andres.figueroa@utrgv.edu

Course Information

Course Website:

Blackboard - <http://my.utrgv.edu/>

All course materials and announcements will be available on Bb

- Syllabus
- Instructor and TA contact information
- Lecture materials
- Assignments, labs, resources, etc
- Due dates, exam schedule, general announcements
- Course announcements and other updates will be handled in class and/OR posted in Bb

Learning Outcomes

Upon successful completion of the course, students will be able to:

- Specify data structures and operations associated with abstract data types.
- Given a scenario, describe the abstract data types that could be created
- Identify, implement, and use the following data structures as appropriate for a given problem:
 - lists, implemented as arrays or linked lists,
 - stacks,
 - queues,
 - binary trees
 - and simple hashes.
- Judge which data model (list, tree, graph, or set) is appropriate for solving a problem
- Justify the choice of a data structure to solve a problem based on issues such as time, space, and of the data structure.
- Judge which sorting algorithms (insertion, selection, merge sort, heap sort, quick sort, radix) is appropriate for solving a problem. Apply the techniques of structured (functional) decomposition to break a program into smaller pieces.
- Describe the mechanics of parameter passing.

This requires a little knowledge and a whole lot of practice

Course Structure

- In class we will:
 - Introduce concepts, commands and explain their use
 - Ask questions
 - Do exercises alone and in groups to find out what you really understood
 - Quick quizzes
 - **Ask more questions**
 - **Repeat as necessary**

Course Structure

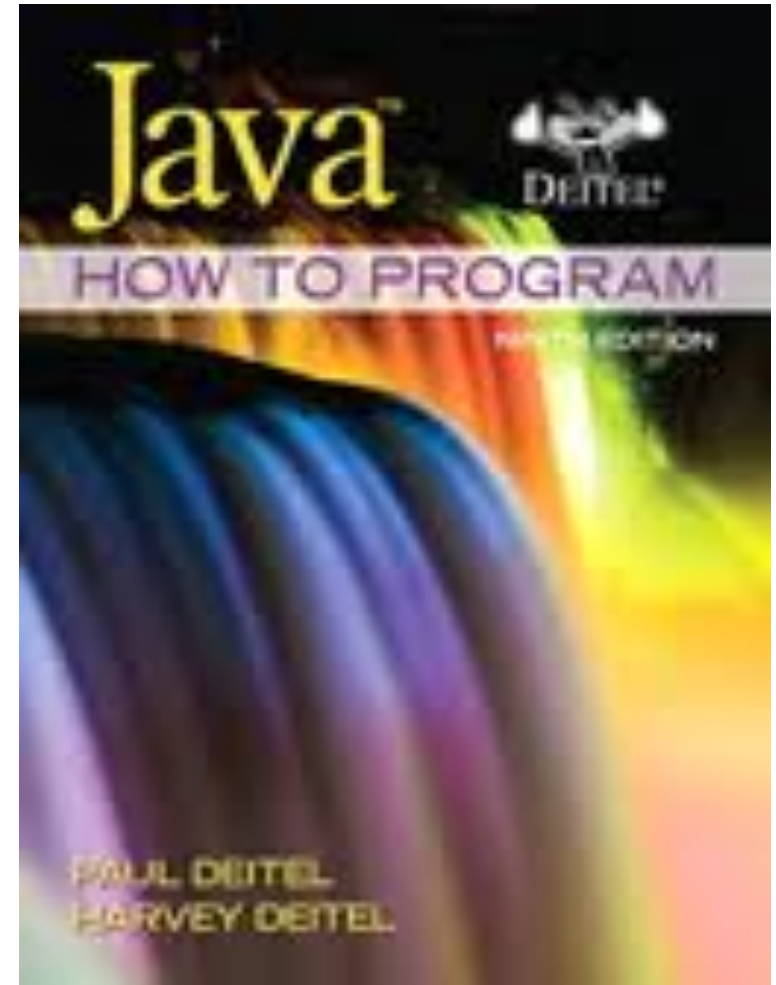
- Outside of class you will:
 - Use your textbook and/or internet resources to review
 - Practice
 - Do review exercises to evaluate your understanding
 - Expect to spend time outside of class doing this every week
 - Programming assignments and projects
 - More practice

Course Structure

- This material is highly cumulative!
 - Once you fall behind, your chances of catching up are not very good
- Material will be presented in 1-2 week chunks
 - Observe, practice, ask, practice, review, practice...
- Each chunk will conclude with an in-class quiz
 - To give you feedback so you know where you stand
 - To give me feedback so I know where we are at

Textbook

Java How to Program, by
Deitel and Deitel, Ninth
Edition, Prentice-Hall, 2012,
ISBN-13: 978-0-13-257566--9.



Resources

- Book resources
 - Video Notes
 - Self-review Exercises with answers
- Online resources
 - Class web site – resources
 - Java SE 7 Documentation
 - Tutorial and demos
- Study guides

Course Requirements

- Students are expected to attend **all** lectures
- In-class exercises and/or quick quizzes
 - Done and turned in during class time
- Programming assignments
- Homework
- Exams
 - Regular quizzes instead of mid-term exam
 - There will be a cumulative final exam, very similar to the regular quizzes

Scoring and Grading

- Weights

Quizzes	25%
Programming Assignments	25%
Homework	25%
Final Test	25%

Total possible score (max):	100%

- Final grade:

90-100%	A
80-89%	B
70-79%	C
60-69%	D
0-59%	F

Late Work

- Exams, quizzes and in-class exercises
 - No make-ups unless arranged beforehand w/ me
- Homework and programming assignments:
 - Late penalty
 - Within 24 hours late will lose 10%
 - Within 48 hours late will lose 20%
 - More than 48 hours late will receive no credit

Questions?