CSE 114, Computer Science 1
Course Information

Spring 2016
Stony Brook University
Instructor: Dr. Paul Fodor

http://www.cs.stonybrook.edu/~cse114
Course Description

• “Procedural and object-oriented programming methodology. Topics include program structure, conditional and iterative programming, procedures, arrays and records, object classes, encapsulation, information hiding, inheritance, polymorphism, file I/O, and exceptions. Software debugging and testing techniques are emphasized. Includes required laboratory.” (https://www.cs.stonybrook.edu/students/Undergraduate-Studies/courses/CSE114)

• Prerequisites: Level 4 or higher on the math placement exam (or MAT 123+)

• Advisory Prerequisite: CSE 101 or ISE108.
General Information

• Meeting Information:
  • Lecture section 1: TuTh 1:00PM - 2:20PM, Javits 110.
  • Lab 1: MoWe 8:30AM - 9:50AM, Computer Sc. 2116.
  • Lab 2: MoWe 10:00AM - 11:20AM, Computer Sc. 2116.
  • Lab 3: MoWe 11:30AM - 12:50PM, Computer Sc. 2116.
  • Lab 4: MoWe 2:30PM - 3:50PM, Computer Sc. 2116.
  • Lab 5: MoWe 4:00PM - 5:20PM, Computer Sc. 2116.
  • Lab 6: MoWe 5:30PM - 6:50PM, Computer Sc. 2116.
  • Lab 7: MoWe 7:00PM - 8:20PM, Computer Sc. 2116.

• Computer Science 2116 is the Computer Science SINC site.
General Information

- **Course Web page:** [http://www.cs.stonybrook.edu/~cse215](http://www.cs.stonybrook.edu/~cse215)
- **Blackboard** will be used for assignments, grades and course material.
- **Staff mailing list:** [cse215ta@cs.stonybrook.edu](mailto:cse215ta@cs.stonybrook.edu)
  - Use this for all communication with the teaching staff
  - Send email to individual instructors only to schedule appointments
Instructor Information

• Dr. Paul Fodor
  214 New Computer Science Building
• Office hours: Tuesdays 11:30AM-1:00PM & Thursdays 2:30PM-4:00PM
• Phone: 1 (631) 632-9820
• Email: paul (dot) fodor (at) stonybrook (dot) edu
  • Please include “CSE 114” in the email subject and your name in your email correspondence
• TAs: see course Web page:
  http://www.cs.stonybrook.edu/~cse114
Textbook

• Intro. To Java Programming, Brief Vers. , Author: Liang , Publisher: Pearson , Edition: 10th, 2014.
  

• Necessary Software:
  
• Eclipse IDE: http://www.eclipse.org

• Important Dates:
  
• Midterm exam 1: Thursday, 3/03/2016, class time, in classroom.
• Midterm exam 2: Thursday, 4/14/2016, class time, in classroom.
• Final exam Lecture section 1: Monday May 16, 2016, 5:30-7:30 PM, Javits 110.
  • http://www.stonybrook.edu/registrar/finals.shtml

• The exams will be like the problems that we solve in the class!
Course Focus

- Introduction to programming (in Java): conditional statements, loops
- Introduce the basic concepts of object-oriented programming
  - object classes
  - encapsulation
  - inheritance
  - polymorphism
  - Application: GUIs
- Fundamental data structures of high-level programming: arrays, lists, stacks, ...
- Algorithms
- Programming assignments
  - systematic software debugging techniques
  - systematic testing techniques
Major Course topics

1. Procedural Programming Basics:
   • variable declarations
   • data types
   • assignment statements & expressions
   • textual manipulation & strings
   • input/output
   • method construction
   • conditional (branching) statements
   • iteration = loops and methods
2. Arrays:
   - collect data in arrays
   - searching
   - sorting
   - array manipulations
3. Object Oriented Programming:
   - designing and constructing classes using containment
   - aggregation
   - inheritance
   - polymorphism
   - Application: GUIs
Coursework

- Grading Schema:
  - Homework, project, quizzes and labs = 25%
    - Programming homework assignments
    - Project
    - Class quizzes
    - Lab assignments
  - Midterm exams (2) = 40% (20% each)
  - Final exam = 35%
Assignments

• Homework assignments due on fixed dates and times.
  • no late submission is permitted
• All assignments should be submitted electronically
  • Blackboard and textbook Web site
Lab exercises

- Simple Coding Exercises done in Computer Science (CS) SINC-site room CS 2116
  - You have only the lab-hour to edit, compile and execute your solution
  - Attendance is mandatory, if you want credit
  - Demonstrate your work to Lab-TA before you leave
- 0 – 3 points:
  - 0 - Student did not attend the lab or program does not even compile.
  - 1 - Student attended the lab, program compiles but has major problems.
  - 2 - Student attended the lab, and program partially works (with some minor errors)
  - 3 - Student attended the lab, and program is correct
Regrading of Homework/Exams

- Please meet with a TA or the instructor and arrange for regrading.
- **You have one week from the day grades are posted or mailed or announced**
- **Late requests will not be entertained**
Academic Integrity

- You can discuss general assignment concepts with other students: explaining how to use systems or tools and helping others with high-level design issues

- You **MAY NOT share** assignments, source code or other answers by copying, retyping, looking at, or supplying a file
  
  - Assignments are subject to manual and automated similarity checking (We do check! and our tools for doing this are much better than cheaters think)

- If you cheat, you will be brought up on academic dishonesty charges - we follow the university policy:
  
  - [http://www.stonybrook.edu/uaa/academicjudiciary](http://www.stonybrook.edu/uaa/academicjudiciary)
Disability

• If you have a physical, psychological, medical or learning disability, contact the DSS office at Room 128 ECC. Phone 632-6748/TDD

• If you are planning to take an exam at DSS office, you need to tell me ahead of time for every exam.

• All documentation of disability is confidential.
Catastrophic events

- Major illness, death in family, ...
- Formulate a plan (with your CEAS academic advisor) to get back on track
- Advice
  - Once you start running late, it’s really hard to catch up
What do you need to get started?

- Blackboard account
  - [http://blackboard.stonybrook.edu](http://blackboard.stonybrook.edu)


- Java:
  - [http://www.oracle.com/technetwork/java/javase/downloads](http://www.oracle.com/technetwork/java/javase/downloads)

- Eclipse IDE:

- Learn to use the debugger!!!

- Liang’s student Web site:
  - [http://www.cs.armstrong.edu/liang/intro10e](http://www.cs.armstrong.edu/liang/intro10e)
Past CSE114 Projects: Artificial Intelligence Poker Spring 2012
Fall 2012 CSE114 Project: Artificial Intelligence Blackjack

User - purse=20; sum=19

Computer - sum=4

Round 1  Action: select bet

Bet: $1  $2  Bet: $0

Action:  Stay  Hit

Result:  => New purse: $  Next

(c) Paul Fodor (CS Stony Brook)
Fall 2013 CSE114 Project: Baccarat
Spring 2014 Project: Pai Gow Poker (double-hand poker)
Fall 2014 Project: 24 Game
Spring 2015 Project: Bridge
Summer 2015 Project: Go Fish

(c) Paul Fodor (CS Stony Brook)
Fall 2015 Project: Wheel of Fortune

(c) Paul Fodor (CS Stony Brook)
Summer 2016 Project: Cosmic Wimpout

http://cosmicwimpout.com/p/7/How-to-play
Tools for Writing Java Programs

• 1\textsuperscript{st} Approach – the bare minimum
  • edit Java source code in text editor (ex: Notepad or Pico)
  • compile source code into class files from command line: javac
  • can be tedious
  • poor interactivity

• 2\textsuperscript{nd} Approach – Integrated Development Environment (IDE)
  • combines writing, compiling, running and debugging Java code into a single application
  • Eclipse, NetBeans, etc.
  • makes coding much more efficient and organized
How does it work?

- Java Source Code
  - you write this
  - `?????.java` files
- Compiler Program
  - `javac ?????.java`
    OR
  - Build - included in the Eclipse IDE
- Result: Java Executable Code
  - `?????.class` files = Java bytecode - not humanly readable
- Java Virtual Machine – runs Java programs
  - `java ProgramName`
    OR Run button included in the Eclipse IDE
Please

- Please be on time
- Please show respect for your classmates
- Please turn off (or use vibrate for) your cellphones

... 

- On-topic questions are welcome
Welcome and Enjoy!