Computer Programming in Java - Fall 2020/Spring 2021

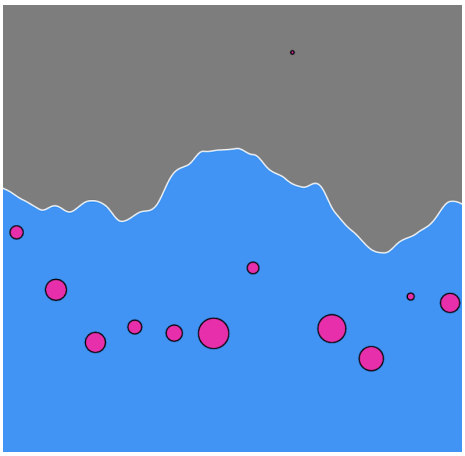
Syllabus

Teacher Chandru Narayan

Room **Remote / Hybrid Benaroya Green**

E-mail chandru.narayan@bush.edu

Course Description
Desired Outcomes
21st Century Learning Expectations
Essential Questions for APCSA exam
Course Materials
Classroom Expectations
Coursework and Grading
Office Hours
Syllabus
Attribution



Course Description

This course is designed to introduce computer programming in the Java language. Learning to use a computer language is a necessary skill for all students regardless of discipline. In this course we will teach the fundamentals of computer programming from the stand point of simulation, automation, and problem solving of real-world systems and natural processes. At the same time, the design and implementation of computer programs is taught from the context of fundamental aspects of computer science, including the development and analysis of algorithms, the development and use of fundamental data structures, the study of standard algorithms and typical applications, and the use of logic and formal methods.

In addition, the year-long course will cover many of the topics necessary for preparation to the AP Computer Science A examination in Spring of the following year. This is an introductory course in computer programming using Java. As such, no specific programming prerequisites are needed to take this course. However, additional preparation may be needed to fully prepare a student for the AP CSA exam with no prior knowledge of computer programming.

Desired Outcomes

Students should be able to at the end of the year-long course:

- Design and implement solutions to problems by writing, running, and debugging computer programs.
- Code fluently in an object-oriented paradigm using the programming language Java.
- Use and implement commonly used algorithms and data structures.
- Read and understand a large program consisting of several classes and interacting objects. (an example of such a program is the AP Computer Science Case Study.)
- Recognize the ethical and social implications of computer use.

21st Century Learning Expectations

Students will create web applications (solutions) not just limited to learning the Java language

- Visual and Project-based learning producing tangible results
- Learn to simulate natural processes and systems through code
- Foundations for Machine learning and Artificial Intelligence
- Develop web-based computer games like Super Mario, Asteroids
- Learn and code advanced Math and Physics concepts
- Develop computer web-based dynamic digital artwork

Students will develop basic computing skills regardless of their discipline of study:

- Engaging in inquiry based, self-driven, authentic learning
- Utilizing higher order thinking skills that include the evaluation and synthesis of information and ideas
- Collaborating and communicating with others while also self-reflecting on personal growth
- Applying appropriate technology and media literacy skills
- Making connections to life and the larger world community

Essential Questions for the student taking the AP CSA exam

- How are Java classes designed to take advantage of reusing existing classes?
- How are arrays of objects and their methods handled in an ArrayList?
- When is it useful to use a wrapper class and the Comparable interface?
- When is it efficient to use a recursive method vs. an interactive method
- Design considerations for managing large amounts of data

Course Materials

Tools/Software:

- Bush Portal
- Bush CPJava course website
- Github & HTML
- Google Classroom Google Drive
- CPJava Slack Channel

Texts/Online Resources:

- Runestone CSAwesome Curriculum
- BuildingJavaPrograms: A Back to Basics Approach by Stuart Reges and Marty Stepp
- repl.it and trinket.io web-based IDE
- Processing IDE and Visual Studio Code IDE
- The Coding Train Youtube Channel

Supplies

- Laptop capable of running local applications
- Robust wireless connection when learning Remotely

Classroom Expectations

- Bring all course materials to class everyday.
- Arrive to class on time and be ready to begin immediately.
- Absences:
    - If you have a planned absence from school (such as commons duty, sport's game, or doctor's appointment), you should inform me before the absence.
    - For any absence (planned or unplanned), you are responsible for checking the portal and asking a classmate about missed information ideally prior to the next class meeting.
- This class will have you in charge of managing a lot of your own work. You need to be an independent worker!
- Take care of other business outside of the classroom and expect to be working for the full period. You will be working on your laptop regularly and with your classmates. Class time is not a time to check e-mail, peruse social media, etc. Don't cheat yourself of valuable class time to ask questions and work with your peers.
- Working together is a great way to learn computer programming. However, if you find that working with peers is less like collaboration and more like appropriation, please talk to me so that we can make a plan so that you can do your best work.

Coursework Grading and Office Hours

Projects and weekly exercises: The majority of class time will be spent completing problem sets to practice and reinforce a computing mindset. We will apply the computing skills we gain to multiple Projects to analyze, document and publish the results. A small set of simple programming tasks with accompanying online exercises will be due approximately once a week. It is your responsibility to use class time to successfully complete and submit code solutions. Then, as time allows, you should challenge yourself to add optional features. You have two additional class periods after the problem set is submitted with comments to make corrections and earn full credit. I will be available during conference hours and at other times by prior appointment to offer any help. Below is the problem set rubric:

3: Mastering	**2: Approaching**	**1: Emerging**	**0: Nothing to assess**
You solved the problem perfectly.	You're on the right track, but made a tiny error or introduced a small bug.	You tried something, but started off on the wrong track or ended up with an invalid answer.	You didn't start the assignment, or you did so little as to be of no benefit to understanding.

Readings/Assignments: In general, your assigned homework will be online readings followed by developing and submitting code extracts. When reading contemporary science articles, you will be required to prepare a response to one or two related prompts which we will discuss the following class period.

Grading

Term grades will be determined by the following:

Readings/Assignments	40%
Projects	40%
Student Portfolio	20%

Class Hours (TBD)

~~Monday~~	~~1:40 - 3:10pm~~	~~Instruction Day~~
~~Wednesday~~	~~9:30 - 10:10am~~	~~Review/Workshop Day~~
~~Thursday~~	~~11:10 - 12:40pm~~	~~Instruction Day~~

Office Hours (TBD)

~~I encourage you to see me outside of class. I will be available in Wisner BANC for conference at the following times. I am happy to make arrangements to meet at other times if needed.~~

~~Monday thru Thursday~~	~~1:20 - 1:40 PM~~
~~Monday and Wednesday~~	~~3:10 - 3:30 PM~~

Syllabus

Table 1-1

LESSON UNITS	APPROXIMATE DURATION	TOPICS
FALL 2020 SEMESTER		
Unit 1	2 Weeks	Introduction to Java, Tools walkthrough, Classroom processes, Environment setup, Java Primitive types and Operators
Unit 2	2 Weeks	Objects, Methods, String, Integer, Double, Math
Unit 3	2 Weeks	Control flow, Booleans, If statements, Object traversals, String, Integer, Double, Math
Unit 4	3-Weeks	Iteration, While loops, For loops, Nested Loops, Loop Analysis
Unit 5	3-Weeks	Anatomy of a class, Constructor, Accessor, Mutator Methods, this keyword, Social Impact of CS
Unit 6	3-Weeks	Arrays, Array Lists, Array Traversal, Final Exam or Project
SPRING 2021 SEMESTER		
Unit 7	2 Weeks	Searching Sorting Algorithms, Ethics of Data Collection, Privacy
Unit 8	2 Weeks	2D Arrays, Traversal
Unit 9	2 Weeks	Projects, Inheritance, Encapsulation, Hierarchy, Polymorphism
Unit 10	2 Weeks	Recursion, Recursive Search. Sort, Final Exam or Project
Unit 11	2 Weeks	Peer Sharing Final Project Presentation

Attribution

The curriculum for this course is adapted from The Coding Train - Dan Shiffman. Lowell High School - Art Simon and Boston Latin School - Long Nguyen