Fall 2018 Updates

Materials and Energy Balances

- New Sections
  - Solver and least squares fits
  - Error and statistics
  - Interpolation
  - 9.9 Integration and numerical integration
  - 9.10 Math functions
  - 9.11 Logical and counting functions
  - 9.12 Sorting and organizing data

Fundamental Programming Concepts

- Full catalog release
- Available for beta in Jan 2018 classes

Data Structure Essentials (Available now)

- New Sections
  - 1.8 Shell sort
  - 1.11 Radix sort
  - 2.10 List header nodes
  - 2.11 Circular lists
  - 3.4 Quadratic probing
  - 3.5 Double hashing
  - 4.10 Heaps using arrays
  - 4.11 Heap sort
  - 5.4 AVL removals
  - 6.10 Topological sort
  - 6.11 Minimum spanning tree
  - 6.12 All pairs shortest path
- Modified sections
  - 3.1 Hash tables (Improvement pass. Added definitions for chaining and open addressing in the collisions subsection)
  - 4.1 Binary trees (New subsection --Special types of binary trees)
  - 5.1 AVL: A balanced tree (New subsection -- Storing height at each AVL node)
  - 5.2 AVL rotations (New subsection -- Algorithms supporting AVL trees, AVL tree balancing)
  - 5.3 AVL insertions (New subsection -- AVL insertion algorithm)
- Miscellaneous
  - The Searching and Sorting Algorithms chapter has been revised to use pseudocode to teach various searching and sorting algorithms, providing a consistent pseudocode approach throughout the zyBook
    - 1.1 Searching and algorithms
    - 1.2 Binary search
    - 1.6 Selection sort
    - 1.7 Insertion sort
    - 1.9 Quicksort
    - 1.10 Merge sort

Circuits (Algebra)

- NEW version of Circuits targeted towards courses that use Algebra instead of Calculus to do analysis

Circuits (Calculus)

- Real world examples added to Calculus
- New section 1.16 Electrical safety

- Improvement pass and additional content in sections:
    - 1.2 Electricity and magnetism
    - 1.3 Current
    - 1.4 Voltage
    - 1.5 Energy and power
    - 1.6 Resistors
    - 1.7 Ohm's Law
    - 1.8 Voltage sources
    - 1.10 Switches
    - 1.12 Capacitors
    - 1.13 Inductors (New participation activities)
    - 1.14 Nonideal voltage sources
    - 2.2 Kirchhoff's Voltage Law (New participation activities)
    - 2.3 Series and parallel combinations
    - 2.4 Voltage divider (New participation activities)
    - 2.5 Kirchhoff's Current Law
    - 2.6 Current divider (New participation activities)
    - 2.7 Maximum power transfer
    - 2.8 Delta-wye (pi-tee) transformations
    - 2.9 Nonlinear resistors
    - 2.10 Practical resistors
    - 2.11 Measuring resistance (New participation activities)
    - 3.2 Node analysis
    - 3.3 Mesh analysis
    - 3.4 Thevenin equivalent (New participation activities)
    - 3.5 Norton equivalent
    - 3.6 Source transformations
    - 4.1 Capacitor constitutive law (New participation activities)
    - 4.2 Inductor constitutive law (New participation activities)
    - 4.3 First-order source-less circuits (New participation activities)
    - 4.4 Singularity functions
    - 4.5 First-order circuit response
    - 4.6 Second-order circuits
    - 4.7 Second-order circuit response
    - 5.1 Dependent sources
    - 5.2 Ideal op-amp
    - 5.3 Feedback
    - 5.4 Inverting amplifier
    - 5.5 Noninverting amplifier
    - 5.6 Integrator and differentiator (New participation activities)
    - 5.7 Practical op-amps
    - 5.8 Multiple op-amp circuits
    - 6.1 Sinusoidal steady state
    - 6.2 Resistor impedance
    - 6.3 Capacitor impedance
    - 6.4 Inductor impedance
    - 6.5 Admittance
    - 6.6 First-order circuit frequency response
    - 6.7 Logarithmic frequency response plots (New participation activities)
    - 6.8 Second order circuit frequency response (New participation activities)
    - 6.9 Practical capacitors and inductors
    - 6.10 Phasors (New participation activities)
    - 6.11 RMS voltages and currents
    - 6.12 Reactive power
    - 6.13 Complex power
    - 6.14 Power factor correction

- New challenge activities in:
    - 2.11 Measuring resistance
    - 3.3 Mesh analysis
    - 3.4 Thevenin equivalent
    - 3.6 Source transformations
    - 5.6 Integrator and differentiator
    - 6.7 Logarithmic frequency response plots

Discrete Math

- Modified sections
    - 1.3 Conditional statements (New end-of-section exercise -- 1.3.5: Translating logical expressions into English)
    - 1.5 Laws of propositional logic (Improvement pass to add steps to proofs and enforce use of commutative law)
    - 1.12 Rules of inference with propositions (Improvement pass to add steps to proofs and enforce use of commutative law)
    - 1.13 Rules of inference with quantifiers (improvement pass)

- New progressions to following section (3 activities with 11 parts)
    - 11.1.1: n-tuples: Permutations and lexicographic order.
    - 11.1.2: r-subsets: Permutations and lexicographic order.
    - 11.3.1: Pigeonhole principle contrapositive.

Linear Algebra

- New catalog release
- Beta version already available for evaluations

MATLAB

- Modified sections/chapter
    - Ch 13 Strings updated to reflect the support of double quotation marks in string declarations
- New section
    - Break and continue
- New / modified challenge activities for the following sections:
    - 5.1 1D element-wise arithmetic operators
    - 6.4 Remainder and modulus functions
    - 6.7 Statistics functions
    - 7.6 Manipulating 2D arrays using a single colon
    - 7.7 Multi-element 2D array indexing using logical arrays
    - 7.9 Indexing 2D arrays using the end keyword
    - 8.7 Sorting
    - 10.2 Relational operators
    - 10.4 Logical operators
    - 10.5 Switch statement
    - 11.4 For loops
    - 11.5 Nested loops
    - 12.1 Scripts with local functions
    - 12.4 Anonymous functions
    - 12.8 Persistent variables
    - 12.9 Recursion
    - 14.1 Grouping data: Structure
    - 14.2 Defining a structure
    - 14.7 Array of structures
    - 15.6 Table
    - 18.2 Curve fitting: Least Squares Regression
    - 18.3 Numerical integration
    - 23.1 Defining Strings and Characters
    - 23.2 Concatenation

    - 23.5 Properties
    - 23.7 Text parsing II
    - TBD

Programming in C

- Modified sections
    - Removed progression -- 1.2.3: Output multiple items with one statement. Earlier CAs cover this material better.
- New progression in the following sections
    - 4.3.1: While loop with sentinel.
- Searching and Sorting chapter moved from DSE
    - 21.1 Searching and algorithms
    - 21.2 Binary search
    - 21.3 O notation
    - 21.4 Algorithm analysis
    - 21.5 Sorting: Introduction
    - 21.6 Selection sort
    - 21.7 Insertion sort
    - 21.8 Quicksort
    - 21.9 Merge sort

Programming in C++

- Modified sections
    - Removed progression -- 1.2.3: Output multiple items with one statement. Earlier CAs cover this material better.
- New Sections
    - 16.8 find () function
    - 16.9 sort () function
    - 15.4 Set (Replaced use of the emplace function with insert function and added an aside)
- New progression in the following sections
    - 4.3.1: While loop with sentinel.
- Searching and Sorting chapter moved from DSE
    - 21.1 Searching and algorithms
    - 21.2 Binary search
    - 21.3 O notation
    - 21.4 Algorithm analysis
    - 21.5 Sorting: Introduction
    - 21.6 Selection sort
    - 21.7 Insertion sort
    - 21.8 Quicksort
    - 21.9 Merge sort

Programming in Java / AP Java / Java EO

- New sections
    - 6.16 Methods and reference
    - 6.17 Returning arrays from methods
    - 6.18 Common errors: Methods and arrays
    - 16.1 Enhanced for loop

- Modified sections
    - Removed progression -- 1.2.3: Output multiple items with one statement. Earlier CAs cover this material better.
    - 8.6 Java example: Employee list using ArrayLists -- Updated example from switch to if/else to remove dependency
    -
- New/modified progressions in the following sections
    - 3.3.2: If-else statements (Converted CA 3.3.2: Multiple if statements: Print car info into a progression with randomization)
    - 3.4.2: Equality and relational expressions (Converted 3.4.2: If-else expression: Detect 100 cents or more. into a progression with randomization)

    - 4.3.1: While loop with sentinel.
    - 4.5.1: For loops (Converted 4.5.1: For loop: Print 1 to N. and 4.5.2: For loop: Print N to 0. into progressions with randomization)
    - 7.5 Objects: Introduction
    - 7.6 Using a class
    - 7.7 Defining a class
    - 7.8 Inline member functions
    - 7.9 Mutators, accessors, and private helpers
    - 7.10 Initialization and constructors
- Searching and Sorting chapter moved from DSE
    - 21.1 Searching and algorithms
    - 21.2 Binary search
    - 21.3 O notation
    - 21.4 Algorithm analysis
    - 21.5 Sorting: Introduction
    - 21.6 Selection sort
    - 21.7 Insertion sort
    - 21.8 Quicksort
    - 21.9 Merge sort

Programming in Python 3

- General improvement pass adds subsection headers to better define subsection topics
- Many sections reorganized or merged
- Modified sections
    - 1.1 Programming (general) (Improvement pass; Combines 1.1 Programming introduction and 1.2 Computational thinking)
    - 1.2 Programming python (Improvement pass; Combines 1.3 Interactive interpreter and 1.4 Programming Python)
    - 1.3 Basic input and output (Improvement pass; Combines 1.5 Output and 1.6 Input)
    - 1.4 Errors (Improvement pass)
    - 1.5 Development environment (Improvement pass)
    - 1.8 Language history (Improvement pass)
    - 2.2 Assignments (Improvement pass)
    - 2.4 Identifiers (was 2.4 Names; Improvement pass)
    - 3.1 String basics (Improvement pass)
    - 3.2 List basics (Improvement pass)
    - 3.3 Dictionary basics (Improvement pass)
    - 3.4 Common data types summary (Improvement pass)
    - 3.6 Type conversions (Improvement pass)
    - 3.7 String formatting (Improvement pass)
    - 3.8 Binary numbers (Improvement pass; Renamed -- was 3.8 "Numbers in binary")
    - 4.2 If-else statement (Improvement pass)
    - 4.3 More if-else (Improvement pass; Renamed -- was 4.3 "Multiple if-else")
    - 4.4 Equality and relational operators (Improvement pass; Renamed -- was 4.4 "Boolean operators and expressions")
    - 4.5 Boolean operators (Improvement pass)
    - 4.6 Membership and identity operators (Renamed -- was 4.5 "Membership operators")
    - 4.9 Conditional expressions (Improvement pass)
- New sections
    - 4.1 If-else branches (general)
    - 4.7 Order of evaluation
    - 16.1 Searching and algorithms
    - 16.2 Binary search
    - 16.3 O notation
    - 16.4 Algorithm analysis
    - 16.5 Sorting: Introduction
    - 16.6 Selection sort
    - 16.7 Insertion sort

    - 16.8 Quicksort
    - 16.9 Merge sort
    - 17.1 zyBooks built-in programming window (Note: this is just a tinker tool that enables students to run code examples)
- New challenge activities in the following sections
    - 3.2 List basics
    - 3.7 String formatting
    - 3.8 Binary numbers
    - 4.4 Equality and relational operators

Programming in Python 2

- General improvement pass adds subsection headers to better define subsection topics
- Many sections reorganized or merged
- Modified sections
    - 1.1 Programming (general) (Improvement pass; Combines 1.1 Programming introduction and 1.2 Computational thinking)
    - 1.2 Programming python (Improvement pass; Combines 1.3 Interactive interpreter and 1.4 Programming Python)
    - 1.3 Basic input and output (Improvement pass; Combines 1.5 Output and 1.6 Input)
    - 1.4 Errors (Improvement pass)
    - 1.5 Development environment (Improvement pass)
    - 1.8 Language history (Improvement pass)
    - 2.2 Assignments (Improvement pass)
    - 2.4 Identifiers (was 2.4 Names; Improvement pass)
    - 3.1 String basics (Improvement pass)
    - 3.2 List basics (Improvement pass)
    - 3.3 Dictionary basics (Improvement pass)
    - 3.4 Common data types summary (Improvement pass)
    - 3.6 Type conversions (Improvement pass)
    - 3.7 String formatting (Improvement pass)
    - 3.8 Binary numbers (Improvement pass; Renamed -- was 3.8 "Numbers in binary")
    - 4.2 If-else statement (Improvement pass)
    - 4.3 More if-else (Improvement pass; Renamed -- was 4.3 "Multiple if-else")
    - 4.4 Equality and relational operators (Improvement pass; Renamed -- was 4.4 "Boolean operators and expressions")
    - 4.5 Boolean operators (Improvement pass)
    - 4.6 Membership and identity operators (Renamed -- was 4.5 "Membership operators")
    - 4.9 Conditional expressions (Improvement pass)
- New sections
    - 1.9 Python example: Salary calculation
    - 4.1 If-else branches (general)
    - 4.7 Order of evaluation
    - 16.1 Searching and algorithms
    - 16.2 Binary search
    - 16.3 O notation
    - 16.4 Algorithm analysis
    - 16.5 Sorting: Introduction
    - 16.6 Selection sort
    - 16.7 Insertion sort
    - 16.8 Quicksort
    - 16.9 Merge sort
    - 17.1 zyBooks built-in programming window (Note: this is just a tinker tool that enables students to run code examples)
- New challenge activities in the following sections
    - 3.2 List basics
    - 3.7 String formatting
    - 3.8 Binary numbers
    - 4.4 Equality and relational operators

Web Programming

- Modified Section
    - Section 11.4 Pug (Renamed -- was 11.4 Jade. Section updated to reflect product name change from Jade to Pug).