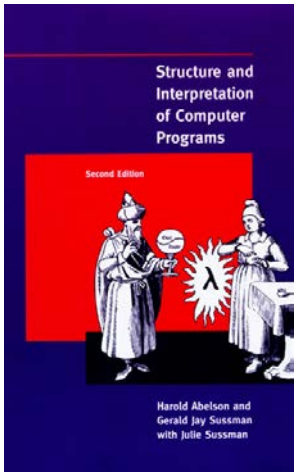
Faculty Review of Open eTextbooks

The California Open Educational Resources Council has designed and implemented a faculty review process of the free and open etextbooks showcased within the California Open Online Library for Education (www.cool4ed.org). Faculty from the California Community Colleges, the California State University, and the University of California were invited to review the selected free and open etextboks using a rubric. Faculty received a stipend for their efforts and funding was provided by the State of California, the William and Flora Hewlett Foundation, and the Bill and Melinda Gates Foundation.

Textbook Name:

Structure and Interpretation of Computer Programs



Find it: eTextbook Website

Textbook Authors:

Harold Abelson and Gerald Jay Sussman with Julie Sussman

Reviewed by:

Fay Zhong

Institution:

California State University, East Bay

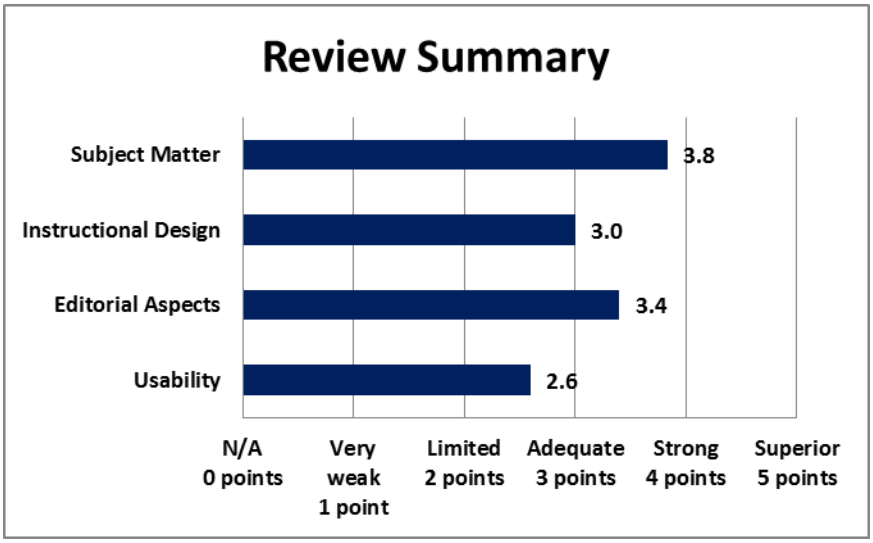
Title/Position:

Professor

Format Reviewed:

Online

A small fee may be associated with various formats.



Date Reviewed:

March 2015

California OER Council eTextbook Evaluation Rubric

CA Course ID: COMP 122

Subject Matter (30 possible points)	N/A (0 pts)	Very Weak (1pt)	Limited (2 pts)	Adequate (3pts)	Strong (4 pts)	Superior (5 pts)
Is the content accurate, error-free, and unbiased?					X	
Does the text adequately cover the designated course with a sufficient degree of depth and scope?					X	
Does the textbook use sufficient and relevant examples to present its subject matter?					X	
Does the textbook use a clear, consistent terminology to present its subject matter?					X	
Does the textbook reflect current knowledge of the subject matter?					X	
Does the textbook present its subject matter in a culturally sensitive manner? (e.g. Is the textbook free of offensive and insensitive examples? Does it include examples that are inclusive of a variety of races, ethnicities, and backgrounds?)				X		

Total Points: 23 out of 30

Please provide comments on any aspect of the subject matter of this textbook:

Instructional Design (35 possible points)	N/A (0 pts)	Very Weak (1pt)	Limited (2 pts)	Adequate (3pts)	Strong (4 pts)	Superior (5 pts)
Does the textbook present its subject materials at appropriate reading levels for undergrad use?				X		
Does the textbook reflect a consideration of different learning styles? (e.g. visual, textual?)					X	
Does the textbook present explicit learning outcomes aligned with the course and curriculum?				X		
Is a coherent organization of the textbook evident to the reader/student?				X		
Does the textbook reflect best practices in the instruction of the designated course?			X			
Does the textbook contain sufficient effective ancillary materials? (e.g. test banks, individual and/or group activities or exercises, pedagogical apparatus, etc.)					X	
Is the textbook searchable?			X			

Total Points: 21 out of 35

Please provide comments on any aspect of the instructional design of this textbook:

Editorial Aspects (25 possible points)	N/A (0 pts)	Very Weak (1pt)	Limited (2 pts)	Adequate (3pts)	Strong (4 pts)	Superior (5 pts)
Is the language of the textbook free of grammatical, spelling, usage, and typographical errors?					X	
Is the textbook written in a clear, engaging style?				X		
Does the textbook adhere to effective principles of design? (e.g. are pages latid0out and organized to be clear and visually engaging and effective? Are colors, font, and typography consistent and unified?)				X		
Does the textbook include conventional editorial features? (e.g. a table of contents, glossary, citations and further references)					X	
How effective are multimedia elements of the textbook? (e.g. graphics, animations, audio)				X		

Total Points: 17 out of 25

Please provide comments on any editorial aspect of this textbook.

Usability (30 possible points)	N/A (0 pts)	Very Weak (1pt)	Limited (2 pts)	Adequate (3pts)	Strong (4 pts)	Superior (5 pts)
Is the textbook compatible with standard and commonly available hardware/software in college/university campus student computer labs?			X			
Is the textbook accessible in a variety of different electronic formats? (e.g. .txt, .pdf, .epub, etc.)			X			
Can the textbook be printed easily?				X		

Does the user interface implicitly inform the reader how to interact with and navigate the textbook?					**X**	
How easily can the textbook be annotated by students and instructors?			**X**			

Total Points: 13 out of 30

Please provide comments on any aspect of access concerning this textbook.

Overall Ratings						
	Not at all (0 pts)	Very Weak (1 pt)	Limited (2 pts)	Adequate (3 pts)	Strong (4 pts)	Superior (5 pts)
What is your overall impression of the textbook?					**X**	
	Not at all (0 pts)	Strong reservations (1 pt)	Limited willingness (2 pts)	Willing (3 pts)	Strongly willing (4 pts)	Enthusiastically willing (5 pts)
How willing would you be to adopt this book?	**X**					

Total Points: 4 out of 10

Overall Comments

If you were to recommend this textbook to colleagues, what merits of the textbook would you highlight?

- This is a well-written hard-core book that covers lots of areas in computer science and programming concepts. It chooses Scheme (a dialect of LISP) as the language for illustration. Scheme is very suitable for this topic because it goes with all the features that support the concepts in the book, for example:

  1) It can pass function as a parameter.

  2) It goes with lambda function (anonymous function).

  3) It has the 'map' and 'apply' methods to embody mapping function to a list or applying a function to a list of objects.

- The book covers many important concepts in Programming Concepts & Methodology, such as tail recursion, iteration vs. recursion, variable scoping (local vs. global, Ch. 3), data abstraction (good preparation for data structure), concurrency and deadlock (for operating systems), lazy evaluation and constraints propagation (lay foundation for Programming Languages class)

- Ch. 4 talks about evaluator that helps in compiler course and logic programming, which will also be discussed in discrete math course.

- Ch. 5 talks about computing with register machine, which is the cornerstone for the learning of assembly language and computer organization and architecture.

- The book has plenty of sample code. Students can under much better with working code. It has lots of pictures/figures, so students can understand the idea or concept with deeper, more concrete impression.

What areas of this textbook require improvement in order for it to be used in your courses?

- One drawback (or challenge) is: this book introduces many details of Scheme; however, Scheme is not a popular language nowadays and thus might make the learning process a bit steep. Some concepts in Scheme such as map, lambda, apply (also appears in other languages like JavaScript), put, amb Evaluator are not easy to grasp.

- Some chapters of the book (for example, 3.3.4 A Simulator for Digital Circuits, 5.1 Designing Register Machines, 5.2 A Register-Machine Simulator, etc.) appear to be less relevant to the basic programming concepts & methodology. Although I agree with the authors that “Designing complex digital systems, such as computers, is an important engineering activity”, computer simulation of circuit design, system design

for performing digital logic simulations, or designing register machines seem to be too complicated for computer science undergraduate students in a lower level programming course.

- More multimedia elements, e.g. animations, audio, video lectures are desired in the textbook.

We invite you to add your feedback on the textbook or the review to the textbook site in MERLOT
(Please register in MERLOT to post your feedback.)

CALIFORNIA OER COUNCIL
CSU + CCC + UC

For questions or more information, contact the CA Open Educational Resources Council.