UNIVERSITY OF CAMBRIDGE COMPUTER LABORATORY

Computer Science Tripos, and
M.Phil in Advanced Computer Science

Selective List of Recommended Reading for Taught Courses

September 2019

This list aims to complement the complete reading lists which are compiled for the M.Phil in Advanced Computer Science, and the Computer Science Tripos. It is intended to satisfy a need expressed by some librarians for more guidance on the most useful books for these degree courses. It is hoped that this list will be particularly helpful to those expanding or planning major changes to their computer science collections.

Very often the lists of "most borrowed" do not give an accurate or complete account of the most useful books. The present list has therefore been compiled based on a number of criteria such as the librarian's experience of the books requested by students, the coverage of course content, and how closely the book's approach matches that of the lecturer.

Priority has been given to those books which are suitable for both the undergraduate and M.Phil courses, but this has not always been possible. In this case, therefore, books have been identified as being suitable for one particular group. Beyond this, the list has been organised into four broad areas of computer science to give a better indication of the applicability of individual titles.

Systems

This topic includes both the design of physical hardware, and the system software necessary for the computer's basic operation. It includes undergraduate courses such as *Operating systems*, *Digital electronics* and *Computer design*. The M.Phil courses *Advanced topics in computer architecture* and *Advanced operating systems* are similar, but go much further. Topics such as databases and computer networking are often covered here.

Gregg, B. & Mauro, J. (2011). *DTrace: dynamic tracing in Oracle Solaris, Mac OS X and FreeBSD*. Prentice Hall. [M.Phil only]

Hennessy, J. & Patterson, D. (2012) *Computer architecture: a quantitative approach.* Elsevier (5th ed.). ISBN 9780123838728

Harris, D.M. & Harris, S.L. (2013). *Digital design and computer architecture.* Morgan Kaufmann (2nd ed.). ISBN 9780123944245 [undergraduate only]

Keshav, S. (1997). *An engineering approach to computer networking.* Addison-Wesley. ISBN 0201634422.

Lemahieu, W., Broucke S. van den & Baesens, B. *Principles of database management.* Cambridge University Press. ISBN 9781107186125 [undergraduate only]

McKusick, M.K., Neville-Neil, G.V., and Watson, R.N.M. (2014). *The Design and implementation of the FreeBSD operating system.* (2nd ed.). Pearson Education. ISBN 9780321968975

Patterson, D. & Hennessy, J. (2017) *Computer organisation and design.* Morgan Kaufmann (RISC-V ed.). ISBN 9780128122761

Peterson, L.L. & Davie, B.S. (2011) *Computer networks: a systems approach.* Morgan Kaufmann (5th ed.). ISBN 9780123850591

Silberschatz, A., Peterson, J.L. & Galvin, P.C. (2014). *Operating systems concepts.* Wiley (9th ed.). ISBN 9781118093757

Programming

This topic covers programming languages, as well as their design and implementation. This includes the popular undergraduate courses like *Programming in C* and *Java.* By the time students reach the M.Phil it is assumed that they will have learned the practical aspects of programming, and courses cover more specialised topics like *Multicore semantics and programming.*

Aho, A.V., Sethi, R. & Ullman, J.D. (2007). *Compilers: principles, techniques and tools.* Addison-Wesley (2nd ed.). ISBN 0321491696

Bratko, I. (2012, 2001). *PROLOG programming for artificial intelligence.* Addison-Wesley (4th, 3rd eds.). [undergraduate only]

Deitel, H.M. & Deitel, P.J. (2017). *Java: How to Program.* Prentice Hall (11th ed.). ISBN 9781292223858 [undergraduate only]

Eckel, B. (2000). *Thinking in C++, Vol. 1: Introduction to Standard C++.* Prentice Hall (2nd ed.). [undergraduate only]

Herlihy, M. & Sahvit, N. (2008). *The art of multiprocessor programming.* Morgan Kaufmann. ISBN 9780123705914 [M.Phil only]

Kernighan, B.W. & Ritchie, D.M. (1988). *The C programming language.* Prentice Hall (2nd ed.). [undergraduate only]

Mitchell, J.C. (2003). *Concepts in programming languages.* Cambridge University Press.

Theory

Theoretical topics cover the application of formal logic and mathematics to provide a rigorous foundation for computer science. This enables the development of precise specifications of hardware and software, while making it possible to reason about systems, or prove them correct. This includes undergraduate courses like *Logic and proof, Types* and *Denotational semantics.* There are also a number of theoretical options for the M.Phil such as *Category theory* and *Formal verification.*

Arora, S. & Barak, B. (2009). *Computational complexity: a modern approach.* Cambridge University Press. ISBN 9780521424264.

Awodey, S. (2010). *Category theory.* Oxford University Press (2nd ed.). [M.Phil only]

Crole, R.L. (1993). *Categories for types.* Cambridge University Press. ISBN 0521450926. [M.Phil only]

Ebbinghaus, H.-D. & Flum, J. (2006). *Finite model theory.* Springer. ISBN 3540287876. [M.Phil only]

Gondran, M. & Minoux, M. (2008). *Graphs, dioids and semirings.* Springer. ISBN 0387754490. [M.Phil only]

Gradel, E., et al. (2007). *Finite model theory and its applications.* Springer. ISBN 9783540004288. [M.Phil only]

Papadimitriou, Ch.H. (1994). *Computational complexity.* Addison-Wesley.

Pierce, B.C. (2002). *Types and programming languages.* MIT Press. ISBN 0262162091

Winskel, G. (1993). *The formal semantics of programming languages.* MIT Press.

Applications

Applications is a broad topic covering practical aspects of computing, and the interface between computer science and other disciplines. Amongst others, this includes the popular courses on machine learning, based on the statistical classification of patterns, and natural language processing. This includes the undergraduate courses *Machine learning and Bayesian inference* and *Formal models of language.* These topics are very popular with M.Phil students too, where there are many similar courses available.

Bishop, C.M. (2006). *Pattern recognition and machine learning.* Springer. ISBN 0387310738

Boreskov, A. & Shikin, E. (2014). *Computer graphics: from pixels to programmable graphics hardware.* CRC Press (2nd ed.). ISBN 9781439867303

Bos, J. & Blackburn, P. (2005). *Representation and Inference for Natural Language and Working with Discourse Representation Theory.* CSLI Press. Available at `http://www.let.rug.nl/bos/comsem/book1.html` [M.Phil only]

Cairns, P. & Cox, A. (2008) *Research methods for human-computer interaction.* Cambridge University Press. ISBN 9780521690317

Goodfellow, I., Bengio, Y. & Courville, A. (2016) *Deep learning.* MIT Press. ISBN 9780262035613 [M.Phil only]

Jurafsky, D. & Martin, J. (2008). *Speech and language processing.* Prentice Hall.

Marschner, S. & Shirley, P. (2016). *Fundamentals of computer graphics.* Peters (4th ed.). ISBN 9781482229394 [undergraduate only]

Poole, D. L. & Mackworth, A. K. (2017). *Artificial intelligence: foundations of computational agents.* Cambridge University Press (2nd ed.). ISBN 9781107195394 [undergraduate only]

Preece, J., Rogers, Y. & Sharp, H. (2015). *Interaction design.* Wiley (4th ed.). ISBN 9781119020752 [undergraduate only]

Russell, S. & Norvig, P. (2010). *Artificial intelligence: a modern approach.* Prentice Hall (3rd ed.). [undergraduate only]