

Massachusetts Institute of Technology  
Department of Electrical Engineering and Computer Science

## Reading List for 1st Lecture

### References

- [Cha95] Craig Chambers. The Cecil language specification and rationale: Version 2.0. Available from <http://www.cs.washington.edu/research/projects/cecil/www/Papers/cecil-spec.html>, December 1995.
- [Cha96] C. Chambers. Synergies between object-oriented programming language design and implementation research. *Lecture Notes in Computer Science*, 1049:250–??, 1996.
- [Coo92] W. R. Cook. Interfaces and specifications for the Smalltalk-80 collection classes. *ACM SIGPLAN Notices*, 27(10):1–15, October 1992. *OOPSLA '92 Proceedings*, Andreas Paepcke (editor).
- [Dij68] E. W. Dijkstra. Goto statement considered harmful. *Communications of the ACM*, 11(3):147–148, May 1968.
- [Hoa89] Hoare. Hints on programming-language design. In *C. A. A. Hoare and C. B. Jones (Ed.), Essays in Computing Science*, Prentice Hall, 1989.
- [Lee91] Peter Lee. *Topics in Advanced Language Implementation*. MIT Press, Cambridge, MA, USA, 1991.
- [NC97] Peter Norvig and David Cohn. Adaptive software. *PC AI Magazine*, January/February 1997.
- [Per82] Alan J. Perlis. Epigrams on programming. *SIGPLAN Notices*, 17(9):7–13, september 1982.
- [Que96] Christian Queinsec. *LISP in small pieces*. Cambridge University Press, Cambridge, UK, 1996.
- [Ste99] Guy L. Steele Jr. Growing a language. *Higher-Order and Symbolic Computation*, 12(3):221–236, October 1999.
- [US91] David Ungar and Randall B. Smith. SELF. the power of simplicity. *Lisp and Symbolic Computation*, 4(3):187–205, July 1991. Preliminary version appeared in *Proc. ACM Symp. on Object-Oriented Programming: Systems, Languages, and Applications*, 1987, 227-241.

- [Wat91] Richard C. Waters. Automatic transformation of series expressions into loops. *ACM Transactions on Programming Languages and Systems*, 13(1):52–98, January 1991.