Reading:
- Mitchell, Ch. 5

Objectives
1. For each of Algol 60, Pascal, C, and ML, be able to describe their historical context, major contributions, and distinctive features.

2. Be able to comprehend, determine the value of, and identify/read the types of simple ML expressions / short programs using basic types, tuples, lists, datatypes, patterns, reference cells, anonymous functions, optional type annotations, pattern matching, and val/fun bindings.

Outline
1. context: the Algol family
   a) family tree
   b) Algol 60
   c) Algol 68 / Algol W
   d) Pascal
   e) C
   f) ML

2. Introduction to ML
   a) Dialects: **Standard ML**, CaML, Objective CaML (OCaML), F#
   b) gcd code
   c) Standard ML of New Jersey
      i. implementation of Standard ML
      ii. compiled
      iii. interactive REPL (read, eval, print, loop) like Lisp/Scheme and Scala
   d) Algol like syntax, but expression oriented
   e) expressions/types
      i. automatic inference, polymorphism (next week)
      ii. no automatic coercion (e.g., between int and real)
iii. basic types: unit, bool, int, string, real
iv. derived types: tuples, records, lists

(f) name binding (val) distinct from:
   i. creating a reference cell (ref)
      A. mutated with :=
      B. accessed with !
   ii. creating an anonymous function (fn)

(g) patterns
   i. case
   ii. val
   iii. fun/fn

(h) user defined constructors (datatype)

Vocabulary
Algod 60, Algol 68, Pascal, ML, Standard ML, CaML, Objective CaML (OCaML), F#, Standard
ML of New Jersey, REPL, unit type, tuple, record, list, constructor, reference cell, dereference,
List const (::)

Further Information
Algod 60 http://www.masswerk.at/algol60/index.htm includes the official reports and
sample code.

ML In addition to the text book, you may wish to examine:

1. a draft of a book by Robert Harper called Programming in ML, which can be found
2. Jeffrey Ullman’s Elements of ML Programming
3. L.C. Paulson’s ML for the Working Programmer (Chapter 9 contains ML code for a
   λ-calculus interpreter)