Lecture 11: Programming in ML

CSC 131
Fall, 2006

Quicksort

fun partition (pivot, nil) = (nil, nil)
| partition (pivot, first :: others) = let
val (smalls, bigs) = partition(pivot, others)
in if first < pivot
then (first :: smalls, bigs)
else (smalls, first :: bigs)
end;

Quicksort - parametrically

fun partition (pivot, nil) lessThan = (nil, nil)
| partition (pivot, first :: others) lessThan = let
val (smalls, bigs) = partition(pivot, others) lessThan
in if (lessThan first pivot)
then (first :: smalls, bigs)
else (smalls, first :: bigs)
end;

Quicksort

fun qsort nil = nil
| qsort [singleton] = [singleton]
| qsort (first :: rest) = let
val (smalls, bigs) = partition(first, rest)
in qsort smalls lessThan @ [first] @ qsort bigs
end;

Defining New Types

- Type abbreviations
  - type point = int * int;
  - type 'a pair = 'a * 'a;
- Datatype definitions
  - create new type with constructors as tags.
  - generative
Datatype Examples

- datatype color = Red | Green | Blue;
- datatype 'a tree = Niltree |

Binary Search Using Trees

fun insert (new:int) Niltree = Maketree(new,Niltree,Niltree) |
insert new (Maketree (root,l,r)) = 
  if new < root 
  then Maketree (root,(insert new l),r) 
  else Maketree (root,l,(insert new r))

fun buildtree [] = Niltree |
buildtree (fst :: rest) = 
insert fst (buildtree rest)

Binary Search Tree

fun find (elt:int) Niltree = false |
find elt (Maketree (root,left,right)) = 
  if elt = root 
  then true 
  else if elt < root 
  then find elt left 
  else find elt right (* elt > root *)

fun bsearch elt list = 
find elt (buildtree list);

Higher-Order Functions

- Functions that take function as parameter
  - Ex: map: (%#a ! #b& ! %%#a list& ! %#b ... 
  - listify oper identity lst defined by:

    fun listify oper identity [] = identity |
    listify oper identity (fst::rest) = oper(fst,listify oper identity rest);

Lazy vs. Eager Evaluation

- Eager: Evaluate operand, substitute operand value in for formal parameter, and evaluate.
- Lazy: Substitute operand in for formal parameter and evaluate body, evaluating operand only when needed.
  - Each actual parameter evaluated either not at all or only once!
  - Like left-most outermost, but more efficient