

CSE 341: Programming Languages

Autumn 2005

Lecture 10 — Mutual Recursion, Equivalence, and Syntactic Sugar

Mutual Recursion

You've already seen how multiple functions can recursively call each other in HW 2.

ML uses the keyword `and` to provide different *scope* rules. Example:

```
fun even i = if i=0 then true  else odd  (i-1)
and odd  i = if i=0 then false else even (i-1)
```

Roughly extends the binding form for functions from `fun f1 x1 = e1` to `fun f1 x1 = e1 and f2 x2 = e2 and ... and fn xn = en`.

Syntax gotcha: Easy to forget that you write `and fi xi = ei`, not `and fun fi xi = ei`.

Mutual Recursion Idioms

1. Encode a state machine (see `product_sign` example)
 - Sometimes easier to understand than explicit state values.
2. Process mutually recursive types, as in HW 2