Pairs and Tuples

Pairs, lists and tuples are way to build up data with multiple parts

What are Pairs

Pairs are things with two parts

We need to build and access them

Building Pair expressions

Syntax: `(e1,e2)` Evaluation: evaluate `e1` to `v1` and `e2` to `v2`, `(v1,v2)` -> that pair of value is a value Type-Check: Theres a new kind of type, if `e1` has type `ta` and `e2` has type `tb` the pair expression has type `ta * tb`

Accessing pair expressions

Syntax: `#1 e` and `#2 e` Evaluation: Evaluate `e` to a *pair of values* and return first or second piece -- if `e` is a variable `x` then look up `x` in the environment Type-check: if `e` has the pair type `ta * tb`, then `#1 e` has type `ta` and `#2 e` has type `tb`

What are Tuples?

Tuples are generalization of pairs, which contain `n` number of expressions.

(e1, e2,..., en) ta,tb,..., tn

Nesting

Pairs and tuples can be nested however you want

```
val x1= (7, (true, 9)) (int * (bool * int)) val x2 = (#1 (#2 x1)) (*bool*)
```