

## CS 365

### Lambda calculus assignment

March 29, 2010

We can encode tuples and extraction of tuples with the Lambda Calculus in the following manner:

$$\begin{aligned} \mathit{Pair} &= \lambda f . \lambda s . \lambda b . ((b f) s) \\ \mathit{Fst} &= \lambda p . (p \mathit{True}) \\ \mathit{Snd} &= \lambda p . (p \mathit{False}) \end{aligned}$$

where *True* and *False* are symbols predefined from class. The way to read the above is that *Pair* takes two arguments (*f* and *s*) and makes an “object” (actually a function, as everything is) which is like an ML tuple containing *f* and *s*. *Fst* and *Snd* are functions that take a pair and return the first and second item, respectively. It’s equivalent to

```
fun Pair(f, s) = (f, s);
fun Fst(p) = #1(p);
fun Snd(p) = #2(p);
```

Confirm that

$$\mathit{Fst} (\mathit{Pair} M N) \rightarrow_* M$$

that is, for expressions *M* and *N*, the expression on the left  $\beta$ -reduces (after one or more steps) to *M*. You may assume, as we showed in class, that *True* *X* *Y*  $\rightarrow_*$  *X*. (Fully parenthesized, the expression on the left is (*Fst* ((*Pair* *M*) *N*)).)