Universität Innsbruck
Institut für Informatik
Prof. Dr. Clemens Ballarin

SS 2008
Exercise Sheet 1
Due 3 April 2008

Automatic Deduction — Introduction to Isabelle

LVA 703522

1 Lambda Calculus

1.1 Church Numerals in Isabelle

In this exercise we will use Isabelle to perform computations with Church numerals. Isabelle's *simplifier*, which is invoked on a proof state by `simp`, performs term rewriting. `unfolding` can be used to unfold definitions.

▷ Make yourself familiar with the `definition` command (Section 4.1.1 of the Isabelle/Isar Reference manual). Use it to define the functions `add`, `mult` and `exp` which perform addition, multiplication and exponentiation, respectively, of Church numerals in Isabelle.

```
definition
  add :: "_" where "add m n = (λf x. n f (m f x))"
definition
  mult :: "_" where "mult m n = (λf x. n (m f) x)"
definition
  exp :: "_" where "exp m n = (λf x. n m f x)"
```

▷ Prove the following lemmas. Which values are computed for `?x`?

```
lemma five: "add (λf x. f (f x)) (λf x. f (f (f x))) = ?x"
  unfolding add_def  — we can unfold the definition first ...
  apply (rule refl)
  done
```

The value `λf x. f (f (f (f x)))` is computed for `?x`.

```
lemma six: "mult (λf x. f (f x)) (λf x. f (f (f x))) = ?x"
  by (simp add: mult_def)  — ... or give it to the simplifier directly
```

The value `λf x. f (f (f (f (f x))))` is computed for `?x`.

▷ Compute $2 + 2 \cdot (3 + 1)$ in a similar fashion.

```
lemma ten: "add (λf x. f (f x))
           (mult (λf x. f (f x))
                 (add (λf x. f (f (f x)))
                      (λf x. f x))) = ?x"
  by (simp add: add_def mult_def)
```

The value λf x. f (f (f (f (f (f (f (f (f x)))))))) is computed for $2 + 2 \cdot (3 + 1)$.

▷ Show that 2 ^ 3 = 2 · 2 · 2.

```
lemma "exp (λf x. f (f x)) (λf x. f (f (f x))) =
       mult (λf x. f (f x)) (mult (λf x. f (f x))(λf x. f (f x)))"
  by (simp add: mult_def exp_def)
```