Isabelle/HOL Exercises
Arithmetic

Magical Methods (Computing with Natural Numbers)

A book about Vedic mathematics describes three methods to make the calculation of squares of natural numbers easier:

- *MM1*: Numbers whose predecessors have squares that are known or can easily be calculated. For example:
  Needed: 61^2
  Given: $60^2 = 3600$
  Observe: $61^2 = 3600 + 60 + 61 = 3721$
- *MM2*: Numbers greater than, but near 100. For example:
  Needed: 102^2
  Let $h = 102 - 100 = 2$, $h^2 = 4$
  Observe: $102^2 = (102 + h)$ shifted two places to the left $+h^2 = 10404$
- *MM3*: Numbers ending in 5. For example:
  Needed: 85^2
  Observe: $85^2 = (8 * 9)$ appended to $25 = 7225$
  Needed: 995^2
  Observe: $995^2 = (99 * 100)$ appended to $25 = 990025$

In this exercise we will show that these methods are not so magical after all!

- Based on *MM1* define a function `sq` that calculates the square of a natural number.

```
consts
  sq :: "nat ⇒ nat"
primrec
  "sq 0 = 0"
  "sq (Suc n) = (sq n) + n + (Suc n)"
```

- Prove the correctness of `sq` (i.e. `sq n = n * n`).

```
theorem MM1[simp]: "sq n = n * n"
  by (induct_tac n, auto)
```

- Formulate and prove the correctness of *MM2*.
  Hints:
  - Generalise *MM2* for an arbitrary constant (instead of 100).
  - Universally quantify all variables other than the induction variable.

```
lemma aux[rule_format]: "!m. m <= n ⟶ sq n = ((n + (n-m))* m) + sq (n-m)"
  apply (induct_tac n, auto)
  apply (case_tac m, auto)
done

theorem MM2:" 100 <= n ⟹  sq n = ((n + (n - 100))* 100) + sq (n - 100)"
  by (rule aux)
```

- Formulate and prove the correctness of *MM3*.
  Hints:
  - Try to formulate the property 'numbers ending in 5' such that it is easy to get to the rest of the number.
  - Proving the binomial formula for $(a+b)^2$ can be of some help.

```
theorem MM3: "sq((10 * n) + 5) = ((n * (Suc n)) * 100) + 25"
  by (auto simp add: add_mult_distrib add_mult_distrib2)
```