Assignment IV (30 pts)

Burak Ekici

Assigned : December the 15th, 19h00
Due : December the 22nd, 23h55

1 Tasks

Q1 (7 pts): Consider an alphabet with two distinct letters A and B inductively implemented in Coq as follows:

```
Inductive letter: Type ≜
  | A: letter
  | B: letter.
```

One can possibly construct in Coq a `type` of `words` using the `letters` in the following fashion:

```
Inductive word: Type ≜
  | Ew: word
  | Cw: letter → word → word.
```

where the constructor `Ew` denotes the "empty word" while the `Cw` is employed to build a `word` out of a given `letter` and a `word` by just prepending the former to the latter.

The `type` of even-length `palindrome words` can also be inductively build in a Coq implementation as below:

```
Inductive pal_even: word → Prop ≜
  | pes : pal_even Ew
  | pnes: ∀ l w, pal_even w → pal_even (Cw l (append w (Cw l Ew))).
```

The task is to prove in Coq that the number of As in an even-length palindrome word is always even:

```
Theorem one: ∀ w, pal_even w → Nat.even (numA w) = true.
```

Hint.
This proof requires structural induction over an instance of the type `pal_even w` to start with, and a case analysis afterwards on the `letter` instance.

Q2 (18 pts): A relation R: A → A → Prop over some type A is said to have the `mirror property` if

$$\forall x,\ y : A,\ x = y \iff \exists\, z : A,\ R\, x\, z\ \wedge\ R\, z\, y.$$

The `mirror property` could simply be reflected in Coq as:

```
Definition mirror {A: Type} (R: A → A → Prop) ≜ ∀ x y, x = y ⟺ ∃ z: A, R x z ∧ R z y.
```

Prove in Coq that

a) (**5 pts**) the relation $RZ := \{x, y : \mathbb{Z} \mid x = -y\}$ over integers has the `mirror property` :

```
Definition RZ: (Z → Z → Prop) ≜ fun x y ⇒ y = Z.mul (-1) x.

Theorem two_a: mirror RZ.
```

b) (**5 pts**) if some relation R over some type A has the `mirror property`, then R is `symmetric`:

```
Theorem two_b: ∀ {A: Type} (R: A → A → Prop), mirror R → (∀ x y, R x y → R y x).
```

c) (**8 pts**) the relation $RN := \{x, y : \mathbb{N} \mid x \neq y\}$ over natural numbers is `symmetric` but does not have the `mirror property`:

```
Definition RN: (nat → nat → Prop) ≜ fun x y ⇒ x ≠ y.

Theorem two_c_one: ∀ x y, RN x y → RN y x.

Theorem two_c_two: mirror RN → False.
```

Nota Bene.
There is no need in principle but you could very well benefit from the lemmata present in the Coq standard library `ZArith` to close the goals in **Q2**.

Q3 (5 pts): Prove in Coq that the Law of Excluded Middle $\forall P, P \vee \neg P$ implies the de Morgan rule $\neg(\exists x : X, \neg P\ x) \implies (\forall x : X, P\ x)$.

```
Definition LEM ≜ ∀ P: Prop, P ∨ ~ P.

Lemma three: ∀ (X: Type) (P: X → Prop), LEM → (~ (∃ x: X, ~ P x) → (∀ x: X, P x)).
```

Important.

Download the accompanying Coq file a4.v from the course DYS page and include your code therein.

2 Submission Policy

Please submit a4.v file. Also, make sure that your code compiles fine with coqc 8.15.

Important Notice.

- Collaboration is strictly and positively prohibited; lowers your score to 0 if detected.
- Any submission after 23h55 on December the 22nd will NOT be accepted. Please beware and respect the deadline!