See the proof for Theorem nat_to_binary_is_zero in table 1

Table 1: Proof of Theorem nat_to_binary_is_zero

Next step in Coq	Proof situation
$Proof.$	——— $forall\ i : nat, binary\_is\_zero\ (nat\_to\_binary\ i) -> i = 0$
$intro\ i.$	$\boxed{\text{i}} : nat$ ——— $binary\_is\_zero\ (nat\_to\_binary\ i) -> i = 0$
$pattern\ i.$	——— $(fun\ n : nat => binary\_is\_zero\ (nat\_to\_binary\ n) -> n = 0)\ i$
$apply\ binary\_induction; [tauto\|\ \|].$	——— $forall\ n : nat, n <> 0 ->(binary\_is\_zero\ (nat\_to\_binary\ n) -> n = 0) ->binary\_is\_zero\ (nat\_to\_binary\ (2 * n)) -> 2 * n = 0$ Hidden 1 goal(s)
$intros\ n\ H1\ IH\ H2.$	$n : nat$ $H1 : n <> 0$ $IH : binary\_is\_zero\ (nat\_to\_binary\ n) -> n = 0$ $H2 : binary\_is\_zero\ (nat\_to\_binary\ (2 * n))$ ——— $2 * n = 0$ Hidden 1 goal(s)
$rewrite\ IH; [tauto\|].$	$n : nat$ $H1 : n <> 0$ $IH : binary\_is\_zero\ (nat\_to\_binary\ n) -> n = 0$ $H2 : binary\_is\_zero\ (nat\_to\_binary\ (2 * n))$ ——— $binary\_is\_zero\ (nat\_to\_binary\ n)$ Hidden 1 goal(s)
$rewrite\ nat\_to\_binary\_even\ in\ H2;\ [\|exact\ H1].$	$n : nat$ $H1 : n <> 0$ $IH : binary\_is\_zero\ (nat\_to\_binary\ n) -> n = 0$ $H2 : binary\_is\_zero\ (false :: nat\_to\_binary\ n)$ ——— $binary\_is\_zero\ (nat\_to\_binary\ n)$ Hidden 1 goal(s)
$inversion\ H2.$	$n : nat$ $H1 : n <> 0$ $IH : binary\_is\_zero\ (nat\_to\_binary\ n) -> n = 0$ $H2 : binary\_is\_zero\ (false :: nat\_to\_binary\ n)$ $x : bool$ $l : list\ bool$ $H3 : false = false$ $H4 : Forall\ (eq\ false)\ (nat\_to\_binary\ n)$ $H : x = false$ $H0 : l = nat\_to\_binary\ n$ ——— $binary\_is\_zero\ (nat\_to\_binary\ n)$ Hidden 1 goal(s)

Continuing proof of Theorem nat_to_binary_is_zero on the next page

Table 1: Proof of Theorem nat_to_binary_is_zero continued

Next step in Coq	Proof situation
assumption.	$\overline{}$ $forall\ n : nat, (binary\_is\_zero\ (nat\_to\_binary\ n)\ {-}{>}\ n = 0)$ ${-}{>} binary\_is\_zero\ (nat\_to\_binary\ (2 * n + 1))\ {-}{>}\ 2 * n + 1 = 0$
intros n IH H1.	$n : nat$ $IH : binary\_is\_zero\ (nat\_to\_binary\ n)\ {-}{>}\ n = 0$ $H1 : binary\_is\_zero\ (nat\_to\_binary\ (2 * n + 1))$ $\overline{}$ $2 * n + 1 = 0$
rewrite nat_to_binary_odd in H1.	$n : nat$ $IH : binary\_is\_zero\ (nat\_to\_binary\ n)\ {-}{>}\ n = 0$ $\boxed{\text{H1}} : binary\_is\_zero\ (true :: nat\_to\_binary\ n)$ $\overline{}$ $2 * n + 1 = 0$
inversion H1.	$n : nat$ $IH : binary\_is\_zero\ (nat\_to\_binary\ n)\ {-}{>}\ n = 0$ $x : bool$ $l : list\ bool$ $\boxed{\text{H2}} : false = true$ $\boxed{\text{H3}} : Forall\ (eq\ false)\ (nat\_to\_binary\ n)$ $\boxed{\text{H}} : x = true$ $H0 : l = nat\_to\_binary\ n$ $\overline{}$ $2 * n + 1 = 0$
discriminate.	Proof finished by Qed

End of proof of Theorem nat_to_binary_is_zero