See the proof for Theorem nat_to_binary_is_zero in table 1

Table 1: Proof of Theorem nat_to_binary_is_zero

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

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

End of proof of Theorem nat_to_binary_is_zero