

Lemma `insere_preserva_ordem`: `forall l x, sorted l -> sorted (insere x l)`.

Proof.

```
induction l. (* A prova é por indução em l. *)
- intro x. (* O caso base é trivial. De fato, seja x número natural. *)
  intro H.
  simpl. (* Precisamos mostrar que a lista contendo apenas x está ordenada, *)
  apply sorted_one. (* mas listas unitárias estão ordenadas por definição. *)
- intros x H. (* No passo indutivo, seja x um natural, e suponha que a lista (a::l) esteja ordenada. *)
  simpl. (* Precisamos mostrar que ao inserirmos x na lista (a::l), obtemos uma lista ordenada. *)
  destruct (x <=? a) eqn:Hle. (* Para isto, precisamos inicialmente comparar x com a. *)
+ apply sorted_all. (* Quando x <= a, concluímos que a lista (x::a::l) está ordenada porque *)
  * assumption. (* x <= a, e *)
  * exact H. (* a lista (a::l) está ordenada. *)
+ generalize dependent l. (* Quando x > a, faremos *)
  intro l. case l. (* uma análise de casos na estrutura da lista l. *)
  * intros IH H. (* Quando l é a lista vazia, *)
    simpl. (* temos que provar que a lista (a::x::nil) está ordenada. *)
    apply sorted_all.
    ** clear IH H. (* Mas isto é verdade já que x > a. *)
      apply Nat.leb_gt in Hle.
      apply Nat.leb_le.
      apply Nat.lt_le_incl.
      assumption.
    ** apply sorted_one.
  * Admitted.
```