```
remove_msgs_disjoint  THM

⊢  ∀A:Type, msg:(A→A→𝔹), L1,L2:A List.
|  (∀a,b:A. (a ∈ L1) ⇒ (b ∈ L2) ⇒ ¬msg(a,b)) ⇒
|  (L1 -msg(a,b) L2) = L1  by Auto THEN Unfold `remove_msgs` 0
|
1. A : Type
2. msg : A→A→𝔹
3. L1 : A List
4. L2 : A List
5. ∀a,b:A. (a ∈ L1) ⇒ (b ∈ L2) ⇒ ¬msg(a,b)
⊢  filter(λa.reduce(λb,y. ¬₂msg(a,b)∧₂y;true₂;L2);L1) = L1
| by BackThruLemma `filter_trivial2` THEN Auto
|
1. A : Type
2. msg : A→A→𝔹
3. L1 : A List
4. L2 : A List
5. ∀a,b:A. (a ∈ L1) ⇒ (b ∈ L2) ⇒ ¬msg(a,b)
⊢  (∀x∈L1.(λa.reduce(λb,y. ¬₂msg(a,b)∧₂y;true₂;L2))(x))
| by Unfolds ``l_all so_apply`` 0 THEN Reduce 0 THEN

|       RWO "l_all_reduce<" 0 THENA Auto
|
1. A : Type
2. msg : A→A→𝔹
3. L1 : A List
4. L2 : A List
5. ∀a,b:A. (a ∈ L1) ⇒ (b ∈ L2) ⇒ ¬msg(a,b)
⊢  ∀x:A. (x ∈ L1) ⇒ (∀b∈L2.¬₂msg(x,b))
| by Unfold `l_all` 0 THEN Auto
|
1. A : Type
2. msg : A→A→𝔹
3. L1 : A List
4. L2 : A List
5. ∀a,b:A. (a ∈ L1) ⇒ (b ∈ L2) ⇒ ¬msg(a,b)
6. x : A
7. (x ∈ L1)
8. b : A
9. (b ∈ L2)
⊢  ¬₂msg(x,b)  by RW assert_pushdownC 0 THENA Auto
|
1. A : Type
2. msg : A→A→𝔹
3. L1 : A List
4. L2 : A List
5. ∀a,b:A. (a ∈ L1) ⇒ (b ∈ L2) ⇒ ¬msg(a,b)
6. x : A
7. (x ∈ L1)
8. b : A
9. (b ∈ L2)
⊢  ¬msg(x,b)  by Easy
END
```