```
member_mapoutl  THM

⊢  ∀A,B:Type, s:(A+B) List, x:A.
   (x ∈ mapoutl(s)) ⟺ (∃y:A+B. (y ∈ s) & isl(y) & x = outl(y))
  by ((UnivCD THEN (Unfold `mapoutl` 0)
       THEN
       (RWO
        Thm* ∀P:(T→𝔹), T':Type, f:({x:T| P(x) }→T'), L:T List, x:T'.
             (x ∈ mapfilter(f;P;L))
             ⟺
             (∃y:T. (y ∈ L) & P(y) & x = f(y))
        {member_map_filter}
        0))
      THENA
      (Auto THEN (All Reduce) THEN (Try ((Analyze -1) THEN Unhide))))
     THEN
     (Reduce 0)
END
```