```
1) Irduktion über Länge den Liste X.
    Def: new 1 (reo1 × Y) = new 1 (rev 1 [] Y) Z

Def: new 1 Y Z = new 1 Y (app [] Z) = new 1 Y (app x Z)
    J-Sarih = X=XS -> X= REEXS
   J. Sellus real (real (REEXS) y) Z= real (rev1 XS (REEY)) Z
         S. Am. new 1 (h: xy) (app xs z) Det. Mev 1 y (h: app xs z) Pet. rev 1 y (app xz)
2) Induktion üler Länge der Linke X
   I And: New (New []) Pet New [] Det. []
   S. Schitt: X=XS -> X= R=3XS
                                                Anmerkung: Da im I-Schritt die I-Annahme nie
                                                benötigt wurde, ist die Aufgabe 2) auch über eine
  J. Solless:
                                                Fallunterscheidung x=[] und x=h::xs lösbar
        rev (rev (Risks)) Defrev (apr (rev xs) [RJ) = rev (rev 1 xs [RJ))

Def. rev (rev 1 (Risks) [J) = app (rev (rev 1 (Risks) CJ) [J=rev 1 (rev 1 (Risks) [J) [J]
         12 rev1 [] (app (h==xs) [] (h==xs) Det. h==xs =x
3) Induktionüber Läng der Liste X
   J.Anf.: reV(app Gev [J) (rev y)) = rev(app [J (rev y)))

oet rev (rev y) = famous

y = app y [J
  1. Shuffi X=XS -> X= Book XS
  5. Sollusos per (app (rev (hesxs)) (rev y) = rev (rev 1 (hesxs) (rev y))
        Defirev (rev 7 XS (hes rev y)) = rev (app (rev XS) (hes revy))
       22 rev lapp (revxs) (rev(rev(harrer y))))
     = app (rev (herrevy)) XS
     Lemnaz
= rev 1 (heerevy) xs= rev1 (revy) (heexs)
      lemaz app (rev (rev V)) (BEEXS) = app y X
```