

## **rel\_1**<sup>13,42</sup>

COM: rel\_1\_begin

COM: rel\_1\_summary

COM: rel\_1\_intro

COM: binrel\_com

ABS:  $\text{Ref}(T;x,y.E(x;y))$  **refl**

STM: refl\_wf

STM: refl\_functionality\_wrt\_iff

ABS:  $\text{Sym}(T;x,y.E(x;y))$  **sym**

STM: sym\_wf

STM: sym\_functionality\_wrt\_iff

ABS:  $\text{Trans}(T;x,y.E(x;y))$  **trans**

STM: trans\_wf

STM: trans\_functionality\_wrt\_iff

STM: trans\_rel\_self\_functionality

ABS:  $\text{EquivRel}(T;x,y.E(x;y))$  **equiv\_rel**

STM: equiv\_rel\_wf

STM: equiv\_rel\_subtyping

ABS:  $\text{Preorder}(T;x,y.R(x;y))$  **preorder**

STM: preorder\_wf

COM: symmetrize\_com

ABS:  $\text{Symmetrize}(x,y.R(x;y);a;b)$  **symmetrize**

STM: symmetrize\_wf

STM: symmetrized\_preorder

STM: trans\_rel\_func\_wrt\_sym\_self

STM: equiv\_rel\_iff

STM: equiv\_rel\_functionality\_wrt\_iff  
 COM: equiv\_rel\_self\_fun\_com  
 STM: equiv\_rel\_self\_functionality  
 STM: squash\_thru\_equiv\_rel  
 ABS: IsEqFun( $T;eq$ ) **eqfun\_p**  
 STM: eqfun\_p\_wf  
 STM: sq\_stable\_\_eqfun\_p  
 ABS: AntiSym( $T;x,y.R(x;y)$ ) **anti\_sym**  
 STM: anti\_sym\_wf  
 STM: anti\_sym\_functionality\_wrt\_iff  
 ABS: StAntiSym( $T;x,y.R(x;y)$ ) **st\_anti\_sym**  
 STM: st\_anti\_sym\_wf  
 ABS: strict\_part( $x,y.R(x;y);a;b$ ) **strict\_part**  
 STM: strict\_part\_wf  
 STM: strict\_part\_irrefl  
 ABS: Connex( $T;x,y.R(x;y)$ ) **connex**  
 STM: connex\_wf  
 STM: connex\_functionality\_wrt\_iff  
 STM: connex\_functionality\_wrt\_implies  
 STM: connex\_iff\_trichot  
 ABS: Order( $T;x,y.R(x;y)$ ) **order**  
 STM: order\_wf  
 STM: order\_functionality\_wrt\_iff  
 ABS: Linorder( $T;x,y.R(x;y)$ ) **linorder**  
 STM: linorder\_wf  
 STM: linorder\_functionality\_wrt\_iff  
 STM: sq\_stable\_\_refl

STM: sq\_stable\_\_sym

STM: sq\_stable\_\_trans

STM: sq\_stable\_\_anti\_sym

STM: sq\_stable\_\_connex

STM: order\_split

ABS: Irrefl( $T;x,y.E(x;y)$ ) **irrefl**

STM: irrefl\_wf

STM: trans\_imp\_sp\_trans

STM: trans\_imp\_sp\_trans\_a

STM: trans\_imp\_sp\_trans\_b

STM: linorder\_le\_neg

STM: linorder\_lt\_neg

COM: rel\_1\_end