

decidable-equality^{0,22}

ABS: EqDecider(T) **deq**

STM: deq_wf

ABS: eqof(d) **eqof**

STM: eqof_wf

STM: deq_property

STM: eqof_eq_btrue

STM: eqof_equal_btrue

STM: strong-subtype-deq

STM: strong-subtype-deq-subtype

STM: nat-deq-aux

ABS: NatDeq **nat-deq**

STM: nat-deq_wf

STM: atom-deq-aux

ABS: AtomDeq **atom-deq**

STM: atom-deq_wf

STM: atom2-deq-aux

ABS: Atom2Deq **atom2-deq**

STM: atom2-deq_wf

ABS: proddeq($a;b$) **proddeq**

STM: proddeq_wf

STM: proddeq_property

ABS: prod-deq-aux $\{v:l,i:l\}(A;B;a;b)$ **prod-deq-aux**

STM: prod-deq-aux_wf

ABS: prod-deq($A;B;a;b$) **prod-deq**

STM: prod-deq_wf

ABS: $\text{product-deq}(A;B;a;b)$ **product-deq**
 STM: product-deq_wf
 ABS: $\text{sumdeq}(a;b)$ **sumdeq**
 STM: sumdeq_wf
 STM: sumdeq_property
 ABS: $\text{sum-deq-aux}\{v:l,i:l\}(A;B;a;b)$ **sum-deq-aux**
 STM: sum-deq-aux_wf
 ABS: $\text{sum-deq}(A;B;a;b)$ **sum-deq**
 STM: sum-deq_wf
 STM: subtype-deq
 STM: subtype_rel-deq
 ABS: $\text{union-deq}(A;B;a;b)$ **union-deq**
 STM: union-deq_wf
 ABS: $\text{deq-member}(eq;x;L)$ **deq-member**
 STM: deq-member_wf
 STM: assert-deq-member
 ABS: $\text{DS}(A)$ **discrete_struct**
 STM: $\text{discrete_struct_wf}$
 ABS: $\text{dstype}(\text{TypeNames}; d; a)$ **dstype**
 STM: dstype_wf
 STM: $\text{decidable_dstype_equal}$
 ABS: $\text{dsdeq}(d;a)$ **dsdeq**
 STM: dsdeq_wf
 ABS: $\text{dseq}(d;a)$ **dseq**
 STM: dseq_wf
 ABS: $x = y$ **eq_ds**
 STM: eq_ds_wf

STM: ds_property
 ABS: insert($a;L$) **insert**
 STM: insert_wf
 STM: insert_property
 STM: no_repeats_insert
 STM: member_insert
 ABS: l-union($eq;as;bs$) **l-union**
 STM: l-union_wf
 STM: member_union
 STM: no_repeats_union
 ABS: l-union-list($eq;ll$) **l-union-list**
 STM: l-union-list_wf
 STM: member_l-union-list
 STM: no_repeats_union-list
 ABS: remove-repeats($eq;L$) **remove-repeats**
 STM: remove-repeats_wf
 STM: remove-repeats_property
 STM: member_remove-repeats
 ABS: list-diff($eq;as;bs$) **list-diff**
 STM: list-diff_wf
 STM: list-diff_property
 STM: member_list-diff
 ABS: IdDeq **id-deq**
 STM: id-deq_wf
 ABS: $a = b$ **eq_id**
 STM: eq_id_wf
 STM: eq_id_self

STM: assert-eq-id
 STM: decidable__equal_Id
 STM: eq_id_test
 ABS: IdLnkDeq **idlnk-deq**
 STM: idlnk-deq_wf
 ABS: $a = b$ **eq_lnk**
 STM: eq_lnk_wf
 STM: assert-eq-lnk
 STM: decidable__equal_IdLnk
 STM: lconnects-transitive
 STM: decidable__l_member
 ABS: KindDeq **Kind-deq**
 STM: Kind-deq_wf
 ABS: $a = b$ **eq_knd**
 STM: eq_knd_wf
 STM: eq_knd_self
 STM: assert-eq-knd
 STM: decidable__equal_Kind
 ABS: $\text{locl_rcv}\{\text{locl_rcv_compseq_tag_def:ObjectId}\}(tg; l; a)$ **locl_rcv_compseq_tag_def**
 ABS: $\text{rcv_locl}\{\text{rcv_locl_compseq_tag_def:ObjectId}\}(a; tg; l)$ **rcv_locl_compseq_tag_def**
 ABS: $\text{locl_locl}\{\text{locl_locl_compseq_tag_def:ObjectId}\}(b; a)$ **locl_locl_compseq_tag_def**
 ABS: $\text{rcv_rcv}\{\text{rcv_rcv_compseq_tag_def:ObjectId}\}(t'; l'; t; l)$ **rcv_rcv_compseq_tag_def**
 STM: map-concat-filter-lemma1
 STM: map-concat-filter-lemma2
 ABS: StandardDS **standard-ds**
 STM: standard-ds_wf
 ABS: $\text{index}(L;x)$ **l_index**

STM: l_index_wf
STM: select_l_index
STM: l_before_l_index
STM: l_before_l_index_le
ABS: has_src($i;k$) **has_src**
STM: has_src_wf
STM: assert-has_src
ABS: has_loc($k;i$) **has_loc**
STM: has_loc_wf
STM: assert-has_loc
ABS: kind_loc($k;i$) **kind_loc**
STM: kind_loc_wf
STM: dependent-pair-inherence
STM: no_repeats_mu_index