testX_prolog_data.txttestX_java_data.txt
LSet
eq $\Longleftrightarrow$ this=sLSetthisLSets
disj $\Longleftrightarrow$ this$\cap$s=
union $\Longleftrightarrow$ this$\cup$s=q
EqTest.java
test1_prolog_data.txt$_p rolog_d ata.txt$
test1_java_data.txt$_j ava_d ata.txt$
EqTest.java
$_A perto_A perto()Solversolver = newSolver(); LSetA1 = newLSet("B1").ins(newLVar("A1")); LSetA2 = newLSet$
$_A perto_C hiuso()Solversolver = newSolver(); LSetA1 = newLSet("B1").ins(newLVar("A1")); LSetC2 = LSet.em$
$_A perto_V ar()Solversolver = newSolver(); LSetA1 = newLSet("B1").ins(newLVar("A1")); LSetV2 = newLSet("$
$_A perto_V uoto()Solversolver = newSolver(); LSetA1 = newLSet("B1").ins(newLVar("A1")); LSetE2 = LSet.emp$
LSet
$$$ local $$$
test2a_prolog_data.txt$_p rolog_d ata.txt$
test2a_java_data.txt$_j ava_d ata.txt$
$_C hiuso_C hiuso()LSetA = newLSet("A"); LSetB = newLSet("B");$
$$$ local $$$
TODO: [i file test2a prolog data.txt e test2b prolog data.txt sembrano identici: corretto?]
test2b_prolog_data.txt$_p rolog_d ata.txt$
test2b_java_data.txt$_j ava_d ata.txt$
$_V ar_A perto()LSetA = newLSet("A"); LSetB = newLSet("B");$
LSet
in $\Longleftrightarrow$ this$\in$s
nin $\Longleftrightarrow$ this$\notin$s
thisLVar***
LVar
test3_prolog_data.txt$_p rolog_d ata.txt$
myTest
$\Longleftrightarrow$ A$\cup$B=C$\wedge$A$\cup$B$\neq$C
$$$ global $$$
test4_prolog_data.txt$_p rolog_d ata.txt$
test4_java_data.txt$_j ava_d ata.txt$
LSet
diff $\Longleftrightarrow$ q=this$\setminus$s
inters $\Longleftrightarrow$ q=this$\cap$s
subset $\Longleftrightarrow$ this$\subseteq$s
test5_prolog_data.txt$_p rolog_d ata.txt$
LRel
comp $\Longleftrightarrow$ qthiss
$q = \{(x, z) \mid \exists y : (x, y) \in this \ \wedge \ (y, z) \in s\}$
inv $\Longleftrightarrow$ sthis
$s = \{(y, x) \mid (x, y) \in this\}$
id $\Longleftrightarrow$ thisLSeta
$this = \{(x, x) \mid x \in a\}$
LSetLRelLRelLSet
test6_prolog_data.txt$_p rolog_d ata.txt$
test6_java_data.txt$_j ava_d ata.txt$
Id
$_V ar_V ar()Solversolver = newSolver(); LRelL4 = newLRel("L4"); LRelL5 = newLRel("L5"); solver.add(L4.id(L5)$
LSetsize
size $\Longleftrightarrow$ n=|this|
nn123
test7_prolog_data.txt$_p rolog_d ata.txt$
test8_prolog_data.txt$_p rolog_d ata.txt$
$\{c[x] : D \mid F[x] \bullet P[x]\}$
c
cx=$\langle x_1, ..., x_n \rangle$
Dc
F[x]x
P[x]x
P[x]c[x]DF[x]
$\{2x \mid x \in D \ \wedge \ x > 0\}$
$\{x : D \mid x > 0 \bullet 2x\}$
RisLSet
```
public Ris(LObject controlTerm, LSet domain, Constraint filter, LObject pattern)
```
```
IntLSet D = new IntLSet("D");
IntLVar x = new IntLVar("x");
Ris a = new Ris(x, D, x.lt(9), x.mul(2));
```
***
test9_prolog_data.txt$_p rolog_d ata.txt$
test9_java_data.txt$_j ava_d ata.txt$
$_L var_C hiuso_F iltroCustom_P attern1_C hiuso()Solversolver = newSolver(); LVarLV1 = newLVar("LV1"); LSetC2 =$
>>
TODO: dire meglio
>
test10_prolog_data.txt$_p rolog_d ata.txt$
LSet
test11_prolog_data.txt$_p rolog_d ata.txt$
test11_java_data.txt$_j ava_d ata.txt$
$_C hiusoGrossoGround_V uoto()$
LSetFullySpecifiedGround_20
testEq_ChiusoGrossoGround_Vuoto()
LRel
test12_prolog_data.txt$_p rolog_d ata.txt$
times.txt
TODO: [dire meglio; non iniziare con Ad esempio]
times.txt