Alloy cheat sheet

Propositional Logic:

$\neg$	negation	`!`, `not`
$\wedge$	conjunction	`&&`, `and`
$\vee$	disjunction	`||`, `or`
$\rightarrow$	implication	`=>`, `implies`
$\leftrightarrow$	equivalence	`<=>`, `iff`

First Order Logic:

$(\forall x)P[x]$	universal quantifier	`all x : X | P`
$(\exists x)P[x]$	existential quantifier	`some x : X | P`
$(\exists! x)P[x]$	uniqness quantifier	`one x : X | P`
$(\exists_0^1 x)P[x]$	at most one quantifier	`lone x : X | P`
$(\forall x)\neg P[x]$	at most zero quantifier	`no x : X | P`

Set Theory:

$\emptyset, \{\}$	the empty set	`none`
$\{x \in X \mid P[x]\}$	comprehension	`{x : X | P}`
$X \subseteq Y$	set inclusion	`X in Y`
$x \in X$	element, singleton subset	`x : one X`
$x \in \wp(X)$	subset inclusion	`x : set X`
$x \in \wp(X), \|x\| \neq 0$	inclusion, nonempty subset	`x : some X`
$x \in \wp(X), \|x\| \leq 1$	empty or singleton subset	`x : lone X`
$X \times Y$	cartesian product	`X -> Y`
$X \cup Y$	sets union	`X + Y`
$X \cap Y$	sets intersection	`X & Y`
$X \setminus Y$	sets difference	`X - Y`
`dom`, U	entire domain set	`univ`

Relational Algebra:

`id`, I	identity relation	`iden`
$R \subseteq A \times B$	binary relation	`R : A -> B`
$R \bowtie S$	relational join	`R.S`, `S[R]`
R^{-1}	inverse/transpose relation	`~R`
R^+	transitive closure	`^R`
R^*	reflexive-transitive closure	`*R`
$S \lhd R$	domain restriction	`S <: R`
$R \rhd S$	range restriction	`R :> S`
$R \oplus S$	relational override	`R ++ S`

Function Properties:

$f : A \pfun B$	partial function	`f : A -> lone B`
$f : A \to B$	total function	`f : A -> one B`
$f : A \pinj B$	partial injection	`f : A lone -> lone B`
$f : A \rightarrowtail B$	total injection	`f : A lone -> one B`
$f : A \psur B$	partial surjection	`f : A some -> lone B`
$f : A \twoheadrightarrow B$	total surjection	`f : A some -> one B`
$f : A \leftrightarrow B$	bijective function	`f : A one -> one B`

Cardinality & Arithmetic:

$\#X, \lvert X \rvert$	set cardinality	`#X`
$1, 2, 3, \ldots$	integer literals	`1,2,3,...`
$x + y$	integer addition	`x + y`, `add[x,y]`
$x - y$	integer substraction	`x - y`, `sub[x,y]`
$x \geq y, x \leq y$	integer comparisons	`x >= y`, `x =< y`
$x > y, x < y$	strict comparisons	`x > y`, `x < y`

more arithmetic functionality with `open util/integer`:

$x' = x + 1$	successor function	`next[x]`
$x \cdot y$	integer multiplication	`mul[x,y]`
$x \,/\, y$	integer division	`div[x,y]`
x `mod` y	reminder division	`rem[x,y]`

Sequences & Lists:

$s = \langle a_0 a_1 \cdots a_n \rangle$	ordered sequence	`seq s`
$s^\frown t$	concatenation	`s.append[t]`
$s_{(0)} = a_0$	head function	`s.first`
$s_{[1]} = \langle a_1 a_2 \cdots a_n \rangle$	tail function	`s.rest`

more ordered sequeces functionality with `open util/sequence`:

Alloy Language Expresions:

signature	`sig` *name* {}
fields	`sig` *name* {*field* : `S`}
abstract/instance	`abstract sig A {}; sig B,C extends A {}`
domain/subset	`sig A {}; sig S in A {}`
shortcuts	`let n1 = s1.field1, n2 = s2.field2 | {n1 = n2}`
facts	`fact {no p: human | p in p.^(mother) && father = ~child}`
predicates	`pred contains[b: Book, n: Name, d: Addr] {n->d in b.addr}`
functions	`fun lives_at[b: Book, n: Name]: set Addr {b.addr[n]}`
conditional	`fun case[n: Int]: Int {n > 0 => 1 else 0}`
assertion check	`assert nSelfF{no m: Man | m = m.father} check nSelfF for 4`
consistency check	`pred Test[M:machine]{doThis} run Test for 4 but 1 machine`

Further information from: `alloy.mit.edu`