c->count(o)  Number of occurrences of o in c

\(a < b\) 
\(a > b\) 
\(a <= b\) 
\(a >= b\)

\(a + b\) 
\(a - b\) 
\(a * b\) 
\(a / b\)

\(a \mod(b)\) 
\(a \div(b)\) 
\(a.\abs()\) 
\(a.\size()\) 
\(a.\max(b)\) 
\(a.\min(b)\) 
\(a.\round()\) 
\(s1.\concat(s2)\) 
\(s1.\size()\) 
\(s1.\toLower()\) 
\(s1.\toUpper()\) 
\(s1.\substring(s, f)\)

if bool expr then expr
else expr endif

\(c->count(o)\)  Number of occurrences of o in c
\(c->excludes(o)\)  True iff o is not an element of c
\(c->excludesAll(c2)\)  True iff all of c2 are not in c
\(c->includes(o)\)  True iff o is an element of c
\(c->includesAll(c2)\)  True iff all of c2 are in c
\(c->isEmpty()\)  True iff c contains no elements.
\(c->size()\)  number of elements in c
\(c->sum()\)  Addition of all elements in c
\(cl - c2\)  Remove elements in c2 from cl if present
\(c->flatten()\)  Merge collection of collection into ‘flat’ collection.
\(c->excluding(o)\)  Remove all occurrences of o from c.
\(c->inclusing(o)\)  Add o to c.
\(c1->union(c2)\)  Merge collections.
\(c1->intersection(c2)\)  Only elements in both c1 and c2.
\(c1->symmetricDifference(c2)\)  Gives collection of elements in exactly one of c1 or c2.
\(c->asBag()\)  Convert to bag (order is lost)
\(c->asOrderedSet()\)  Convert to ordered set
\(c->asSequence()\)  Convert to sequence.
\(c->asSet()\)  Convert to set.
\(c->append(o)\)  Append to end.
\(c->prepend(o)\)  Insert at beginning.
\(c->at(i)\)  \(i^{th}\) element.
\(c->first()\)  first element.
\(c->last()\)  last element.
\(c->indexOf(o)\)  Index of first occurrence of o (indexed from 1)
\(c->insertAt(i, o)\)  Insert o at index i.
\(c->subOrderedSet(l, u)\)  OrderedSet only.
\(c->subSequence(l, u)\)  Sequence only.
\(c->exists(exp)\)  True iff at least one element in c makes exp true.
\(c->forAll(exp)\)  True iff exp is true for every element in c.
\(c->isUnique(exp)\)  True iff exp has a unique value for every element in c.
\(c->one(exp)\)  True iff there is exactly one element in c for which exp is true.
\(c->any(exp)\)  A random element for which exp is true.
\(c->collect(exp)\)  All objects resulting from exp on elements of c.
\(c->collectNested(exp)\)  Collection of collections resulting from exp on elements of c.
\(c->reject(exp)\)  Subcollection of c containing elements for which exp is false.
\(c->select(exp)\)  Subcollection of c containing elements for which exp is true.
\(c->sortedBy(exp)\)  Ordered Subcollection of c with elements ordered according to increasing exp.

\(a^@pre\)  The value of a at the start of execution of the operation.
\(result\)  The value returned by the operation.
\(v->oclIsNew()\)  True iff v is constructed during execution of the operation.
\(a^@op(arg)\)  \(\text{isSent}:\) True iff the operation has sent (called) op(arg) on a during its execution.
\(a^"\op(arg)\)  \(\text{message operator}:\) The sequence of messages sent that match op(arg) during the execution of the operation.
\(m->hasReturned()\)  True iff m has finished executing
\(m->result()\)  Return value of m.
\(m->isSignalSent()\)  True iff m is a signal.
\(m->isOperationCall()\)  True iff m is an operation call.
\(o->oclIsUndefined()\)  True iff o is undefined.
\(o->oclIsTypeOf(<Type>)\)  True iff o of type <Type>.
\(o->oclIsKindOf(<Type>)\)  True iff ooclIsTypeOf(<Type>) or o is an instance of a subtype of <Type>.
\(o->oclInState(<sname>)\)  True iff o is in the state named <sname>. o must have associated state chart.
\(type::allInstances()\)  The set of all instances of type. (usage discouraged)