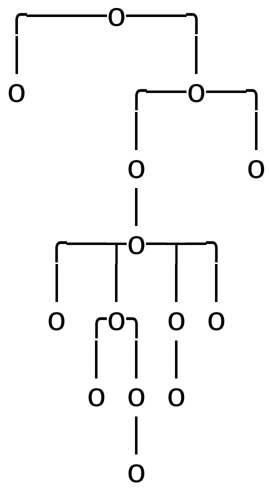
Combinatorics and algebra of partially ordered sets



Partially ordered set = finite set P endowed with a partial order relation $\leq$

abbreviated as "poset" (en français, ordre partiel)

somewhere between

- (A) combinatorics, more precisely graph theory
- (B) algebra, more precisely representation theory

On the (A) side : a poset is an acyclic and transitively-reduced directed graph

On the (B) side : a poset is a finite category in which each set $\mathrm{Hom}(x, y)$ has at most one element

So posets are both **very concrete objects** on which one can look for algorithms solving discrete problems, but also **algebraic entities** that possess a very rich representation theory.

And a lot of fun : many kinds of cool posets appear in nature. Name your favorite !

The directed graphs in (A) (acyclic and transitively-reduced) are called **Hasse diagrams**.

- acyclic means: no oriented cycles
- transitively-reduced means: no configuration of edges $x \to y \to z$ and $x \to z$

From a poset P, its Hasse diagram is made of

- vertices = elements of P
- edges $x \leftarrow y$ = cover relations $x \le y$ in P

(note the visual convention here : arrows go decreasing)

cover relation = pair (x, y) with $x \neq y$ such that $x \le z \le y$ implies $z = x$ or $z = y$.

Backward, to recover the partial order relation from one Hasse diagram H

- elements of P = vertices of H
- $x \leq y$ in P if and only if there is a directed path in H from y to x.

Note : this is like a transitive closure operation on directed graphs.

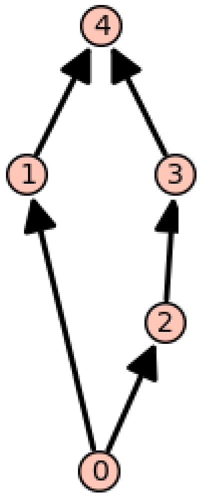
Here is an example of poset, seen through its Hasse diagram.

(all examples in SageMath)

This one has 5 elements, 5 cover relations and 13 relations $x \leq y$.

In [1]:
```
P = posets.PentagonPoset();P.plot(figsize=4)
```

Out[1]:



The category of a poset P (an algebraic viewpoint)

This is a finite category : finite set of objects and finite Hom set between any two objects

denoted by cat(P)

objects = elements of P = vertices of the Hasse diagram of P

Hom(x, y) has either one unique element $\star_{x,y}$ if $x \geq y$ and no element otherwise

(convention here : morphisms go decreasing, follow the arrows)

The axioms of category come from

- reflexivity => existence of Id_x in Hom(x,x)
- transitivity => composition is well-defined

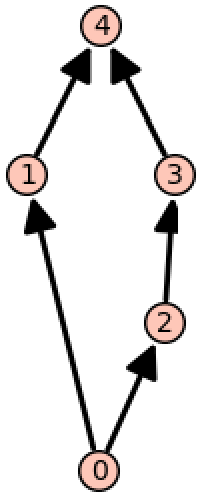
Then associativity and unit axiom are clear, because $|Hom(x, y)| \leq 1$

So for example, $Hom(1,2) = \emptyset$ and $Hom(2,4) = \{\star_{2,4}\}$ in the example below

empty if no path, singleton if at least one path

In [2]:
```
P.plot(figsize=4)
```

Out[2]:



Operations on posets

One can speak of isomorphisms between posets = isomorphisms of Hasse diagrams as directed graphs.

So here is a first question about any two posets

- (Q_0) are P and Q isomorphic ?

Can be seen as a special case of isomorphism between directed graphs

There are easy operations on posets

- dual poset of P = opposite Hasse diagram, returning the partial order and the arrows,
- disjoint union of P and Q, where the empty poset is the neutral element,
- (cartesian) product of P and Q = product of Hasse diagram as directed graphs

partial order on $P \times Q : (x, x') \leq (y, y')$ iff $x \leq x'$ and $y \leq y'$

Any poset with 1 element is a neutral element up to isomorphism for the product.

There are then very natural questions, for a given poset P:

- (Q_1) is the poset P isomorphic to its dual poset (**self-dual**) ?
- (Q_2) is the poset P isomorphic to a cartesian product of smaller posets (**decomposable**)?

All these questions (Q_0), (Q_1), (Q_2) can be solved using general algorithms for directed graphs.

But maybe there are better specific algorithms that apply only to posets and Hasse diagrams ? Can one gain something in complexity ?

Visual display (side remark)

general issue of finding a nice way to display directed graphs, to use our visual brain

one can apply general algorithms for directed graphs (springs, planar layout, projected 3D, etc), for the Hasse diagrams

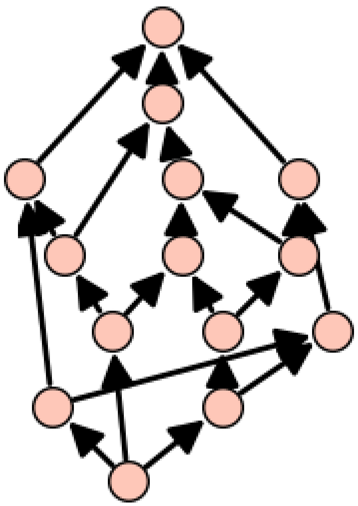
sometimes gives correct results, but not always

not adapted to the visual display of the partial order relation !

unless there is some kind of natural height function and the display preserves this height

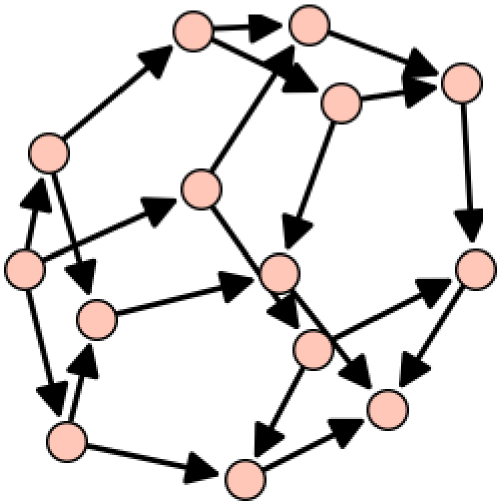
In [3]:
```
posets.TamariLattice(4).plot(label_elements=False,figsize=4)
```

Out[3]:



In [4]:
```
posets.TamariLattice(4).hasse_diagram().plot(vertex_labels=False,figsize=4)
```

Out[4]:



Möbius function

One important use of partial orders is *Möbius inversion*.

Let f and g be two functions from P to some fixed ring R.

Then there is an equivalence between $f(x) = \sum_{x \leq y} g(y)$

and $g(x) = \sum_{x \leq y} \mu(x, y) f(y)$

where μ is a function from $P \times P$ to $\mathbb{Z}$ called the Möbius function.

The function μ is unique, and determined by the property

$\forall x, y \sum_{x \leq z \leq y} \mu(x, z) = \delta_{x,y}$

Example: for the boolean lattice of subsets of a set, $\mu(S, T) = (-1)^{|T \setminus S|}$.

Möbius matrix

One can think of the Möbius inversion as matrix inversion.

In order to make matrices in the usual sense, one must choose an increasing numbering of the elements of the poset P by the integers between 1 and $|P|$. There are usually many such choices.

Then one can encode the partial order relation $x \leq y$ into a triangular matrix

$M_{x,y} = \delta_{x \leq y}$; with 1s on the diagonal

and the Möbius matrix is just the inverse of this matrix. Example:

In [5]:
```
P.lequal_matrix(),P.moebius_function_matrix()
```

Out[5]:
```
(
[1 1 1 1 1]  [ 1 -1 -1  0  1]
[0 1 0 0 1]  [ 0  1  0  0 -1]
[0 0 1 1 1]  [ 0  0  1 -1  0]
[0 0 0 1 1]  [ 0  0  0  1 -1]
[0 0 0 0 1], [ 0  0  0  0  1]
)
```

Incidence algebra

Another way to thing about the Möbius matrix is to use the incidence algebra.

Choose a base ring R, for instance $\mathbb{Q}$ or $\mathbb{Z}$

Let us define an associative algebra $I(P)$.

It has a basis $B_{x,y}$ for every relation $x \leq y$ in P (all relations, not only the cover relations)

One can think of $B_{x,y}$ as any path from y to x in the Hasse diagram, or as the unique morphism from y to x in the category cat(P)

The product $B_{x,y}B_{z,t}$ is either $B_{x,t}$ if $y = z$ or zero otherwise

Think about composable paths in a directed graph

This gives a finite dimensional associative algebra $I(P)$ for every poset P.

The dimension of the incidence algebra $I(P)$ is a basic numerical invariant of a poset, this is the number of comparable pairs of elements. This number is invariant under duality and multiplicative for the Cartesian product of posets.

One can also think of the incidence algebra

as the subalgebra of upper triangular matrices N that have non zero coefficient in position (x, y) only if $x \leq y$.

$$\begin{pmatrix} * & * & * & * & * \\ 0 & * & 0 & 0 & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & 0 & * \end{pmatrix}$$

So the Möbius function is the inverse in $I(P)$ of the element with 1 everywhere

$\sum_{x \leq y} B_{x,y}$

which is sometimes called the zeta element.

The names Möbius and zeta are inspired by the poset $\mathbb{N}$ ordered by divisibility.

Core problem : Computing the Möbius function

- Naive algorithm : create the matrix M, invert this matrix

building on very fast existing algorithms for linear algebra over $\mathbb{Z}$

(and this is in fact a very special inversion, for a sparse triangular matrix)

- Another algorithm : use increasing induction on the poset

This is smarter, but maybe there is a better way still.

More generally, one could ask:

How to compute efficiently in the incidence algebra $I(P)$?

Using matrices does not seem to be a good idea. One should rather use graph-theoretic ideas, and even find poset-specific algorithms.

Hasse diagram and incidence algebra

Yet another viewpoint on the incidence algebras (*quiver viewpoint*)

- R-linear combinations of paths in the Hasse diagram of P

modulo the equivalence relation "*having same start and same end*" between paths

any two paths from x to y are identified modulo this equivalence relation

- product = concatenation of paths

One can associate a representative path from y to x to each basis element $B_{x,y}$ with $x \leq y$.

Modules and representations

$I(P)$ is an associative algebra over the base ring R.

So one can talk about **modules over P**, meaning modules over $I(P)$

The category of finite dimensional modules over P is an Abelian category (kernel, cokernel, short exact sequences).

One can consider either left modules or right modules. This does not make much difference, because left-modules over I(P) amount to right modules over the incidence algebra of the dual poset of P.

Alternative viewpoint = functors from the category cat(P) to the category of R-modules

This last viewpoint allows to define representations with values in any Abelian category, for examples chain complexes, Hodge structures, sheaves, etc.

Note: These categories of modules were already considered in 1975 by Loupias. Some articles on the subject since then, but not so many.

Alternative point of view : representations of the Hasse diagram

let H be the Hasse diagram of a poset P

a representation (V, f) of H over the field R is the data

- of one vector space V_x for every vertex x of H
- of one linear map $f_{x,y}$ from V_x to V_y for every arrow $x \to y$ in H

that must satisfy the following commutations

- for all x, y, all linear maps associated to paths from x to y are equal

Here the linear map of a path is defined by composition of the $f_{x,y}$.

Morphisms are then given by linear maps g_i that make commuting diagrams.

Examples:

simple modules S_x with one non-zero space $(S_x)_x$ and all maps are zero

projective modules P_x with $(P_x)_y = R$ iff $x \geq y$ and identity maps or zero

injective modules I_x with $(I_x)_y = R$ iff $x \leq y$ and identity maps or zero

more generally, for each interval $[x, y]$ and for each *convex* subset K of P, one can associate a representation of P : a copy of the ring R on every element of K and the identity map wherever possible.

Remark:

- every representation has a unique decomposition as a direct sum of indecomposable representations
- in general, infinitely many iso-classes of indecomposables. The classification problem is *wild* and therefore hopeless, with continuous parameters. This happens even for rather simple-looking posets, for example a star-shaped tree with 6 vertices.

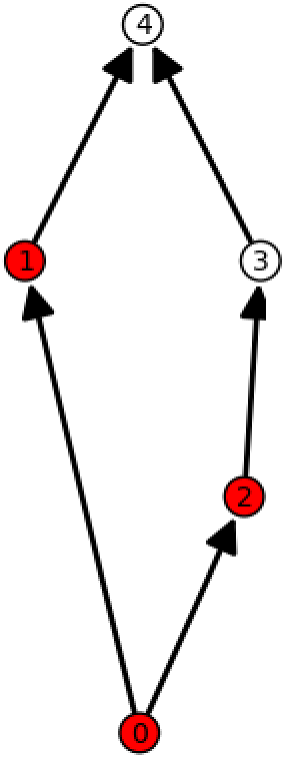
Example of representation: (equality of two paths holds)

- $\mathbb{Q}$ on red vertices
- 0 vector space on white vertices
- Id on arrows between red vertices
- 0 map on other arrows

In [6]:

```
P.plot(element_colors={'red':[0,1,2]},element_color='white')
```

Out[6]:



a little bit of homological algebra

height h of P := length of longest maximal chain $x_1 < x_2 \cdots < x_h$.

Proposition : every representation M of P has a *projective resolution* of length at most h.
(chain complex of projective modules whose homology is P in degree 0)

proof by induction on the height of the support of the module

hence algorithm for finding a projective resolution

implemented in HAP package of GAP

⚠️ Attention : projective resolutions are not unique, but unique up-to-homotopy.

=> each module has a unique representative in the bounded derived category $D^b P$

$D^b P$ is defined as the homotopy category of bounded complexes of projective modules

Take that as a **blackbox** ; think of is as a *simplified* version of the module category ; objects are finite chain complexes of projectives, so this remains rather computable

Derived category as an invariant for posets

Définition: let P and Q posets. P is derived equivalent to Q, denoted $P \simeq^d Q$ (over the ring R) if the derived categories $D^b P$ and $D^b Q$ (over R) are equivalent as triangulated categories.

To decide if $P \simeq^d Q$ is a difficult problem in general.

- to show that $P \not\simeq^d Q$, one can look for invariants depending only on derived categories and taking distinct values => need for refined but computable invariants
- to show that $P \simeq^d Q$, one can search for a tilting-complex between P and Q : an object of $D^b P$ whose endomorphisms recover the incidence algebra of Q (something like an incarnation of Q inside the modules over P)

BUT : 🛑 finding a tilting complex is very hard as the classification of indecomposable objects in $D^b P$ is almost always wild. Like walking at random in hyperbolic space..

One can also look directly for functors giving the equivalence, not easy either

Examples of derived equivalences

Let P and Q be given by Hasse diagram that are orientations of the same unoriented tree ; then P and Q are derived-equivalent. (using Bernstein-Gelfand-Ponomarev reflexions functors) This example is really about quivers, rather specific posets.

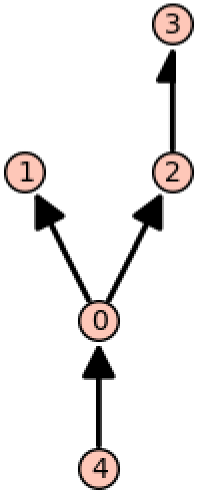
Let P be a poset with unique maximal element t. Let Q obtained from P by removing t, then adding a new minimal element b. Then P is derived-equivalent to Q. (Ladkani)

Particular case of the powerful construction "flip-flop" introduced by Ladkani. Given two posets P, Q and a map from P to Q with some hypotheses, one can define two derived equivalent posets, where P is either below or above Q.

In [7]:

```
Q = Poset(DiGraph(P.subposet([0,1,2,3]).cover_relations() + [[4,0]]));Q.plot(figsize=4)
```

Out[7]:



Basic Invariants of posets

- (A) cardinality of P
- (B) valence bi-variate polynomial : $\sum_{x \in P} a^{i(x)} b^{o(x)}$ where $i(x)$ and $o(x)$ are the incoming and outgoing valences in x in the Hasse diagram
- (C) number of intervals in P, i.e. number of pairs (x, y) such that $x \leq y$

(which is also the dimension of the incidence algebra)

- (D) Zeta polynomial of P, counting k-chains in P for all k

These 4 invariants are multiplicative, hence allow to show that a poset is not isomorphic to a cartesian product of smaller posets. Only (A) is invariant under derived equivalence.

More subtle invariants

Recall the matrix M_P with $M_{x,y} = \delta_{x \leq y}$ encoding the poset P.

Define the **Coxeter matrix** as $C_P = -M_P(M_P^t)^{-1}$.

Proposition : up to conjugation by an invertible matrix over $\mathbb{Z}$, the matrix C_P only depends on the derived category of P.

Not so easy to check if two matrices are conjugate over the integers !

Définition : the **Coxeter polynomial** of P is the characteristic polynomial of C_P.

Our main concrete derived invariant of posets.

In [8]:
```
P.coxeter_transformation(),P.coxeter_polynomial()
```

Out[8]:
```
(
[ 0  0  0  0 -1]
[ 0  0  0  1 -1]
[ 0  1  0  0 -1]
[-1  1  1  0 -1]
[-1  1  0  1 -1], x^5 + x^4 + x + 1
)
```

In [9]:
```
P.coxeter_polynomial() == Q.coxeter_polynomial()
```

Out[9]:
```
True
```

Some comments:

- equality of Coxeter polynomials does not imply derived equivalence

(there are counter-examples, probably many)

- but equality in famillies still gives a strong hint in favor of derived equivalence

Example: Tamari lattices (binary trees 🌳 under rotation) and Dyck paths under inclusion : conjectural derived equivalence, with strong hints

- No known algorithm to decide derived equivalence, could be undecidable ?
- Basic strategy trying to prove that P and Q are derived equivalent :
    - find conjugating matrices over $\mathbb{Z}$ between the Coxeter matrices of P and Q with small entries (by lifting conjugating matrices over small finite fields).
    - lift the nicest among such matrices to a tilting complex and prove that it is indeed a tilting complex.

Comments on implementations

The Coxeter polynomial is rather easy to compute : $C_P = -M_P(M_P^t)^{-1}$ so

- one triangular matrix inversion
- one matrix transposition
- one matrix product (lower triangular times upper triangular)
- one characteristic polynomial (sparse matrix over $\mathbb{Z}$)

Sage implementation takes a reasonable time for posets of size at most $\simeq$ 2000.

Using magma for the last step, one can go a bit further.

Last step seems to be the blocking point.

I do not know of any low-level implementation of the Coxeter polynomial.

Also implemented in Macaulay2 Poset package. Not available in Maple® or Mathematica®.

Derived factorisation of posets

For a cartesian product of posets $P \times Q$, the categories of modules over the incidence algebra and the derived categories are cartesian products of categories.

Therefore the Coxeter matrix of $P \times Q$ is the **tensor product** of those of P and Q.

Problem: Can one recognize if a poset is derived-equivalent to a cartesian product of posets ?

One tool : decide if the Coxeter polynomial is the characteristic polynomial of a tensor product of matrices.

This could be done by factorising the Coxeter polynomials, but probably not the best way to do it. Any better algorithm ?

⚠️ not the usual kind of factorisation for polynomials ! set of roots must be a Cartesian product !

$\prod_{i,j}(z - r_i r'_j)$ from $\prod_i (z - r_i)$ and $\prod_j (z - r'_j)$

The Coxeter functor (where does all this comes from)

The Coxeter matrix comes from a functor from $D^b P$ to itself, called the Coxeter functor or the Auslander-Reiten translation functor τ. Also closely related to the Serre duality functor.

The Auslander-Reiten theory is a very central tool in modern representation theory of associative algebras.

At the level of objects, the functor τ maps the indecomposable projective object P_x to the shifted indecomposable object $I_x[1]$.

This is therefore something that can be implemented easily, and that is available in HAP

The functor τ acts linearly on the K_0 of the category $D^b P$: this is the Coxeter matrix C_P that was introduced before.

Fractionally Calabi-Yau posets 🍰

now introducing a *very interesting* and special sub-class of posets.

A poset P is called *fractionally-Calabi-Yau* if some power of τ is isomorphic to some power of the shift functor [1] in $D^b P$ (which just shift the indices in chain complexes).

Definition by Kontsevich, comes from the geometry of Calabi-Yau varieties, mirror symmetry, algebraic geometry, etc

This condition implies that the Coxeter matrix has a power that is $\pm$ the identity matrix : indeed $\tau^p = [q]$ implies $C_P^p = (-1)^q$.

And therefore, the Coxeter polynomial is a product of cyclotomic polynomials.

Remark: for an arbitrary poset, roots of the Coxeter polynomial are arbitrary

In [10]:
```
posets.DiamondPoset(6).coxeter_polynomial().complex_roots()
```

Out[10]:
```
[-1.00000000000000, 0.381966011250105, 2.61803398874989]
```

PROBLEM : How to prove that a poset is fCY ?

one needs to understand the functor τ on sufficiently many objects (not so easy)

Much easier sometimes to prove the weaker statement that the roots of the Coxeter polynomial are on the unit circle

For example, both statements known for the *Tamari lattices*

posets : planar binary trees with n leaves ; Hasse diagram = rotation of binary trees

roots of Coxeter polynomial described using operad theory and Koszul duality ;

more recently, Rognerud proved the fCY property

Techniques are very specific to this particular family of posets.

Many examples of nice posets conjectured to be fCY : a lot of open problems ! includes plane partitions, m-Dyck paths, minuscule posets, etc

Bonus track 🥧

The monoid of Weight Symbols

This is a tool used in the study of fractionnally Calabi-Yau posets, and their factors. This provides an enriched version of the Coxeter polynomial.

A *weight symbol* is a pair $([e_1, \ldots, e_n], N)$ where e_i are positive integers (exponents) and N is a positive integer (weight, larger than any e_i)

For example $([3, 4, 5], 20)$. The order among the e_i is irrelevant.

The product of (e,N) and (e',N') is defined as follows: let M be the lcm of N and N'.

Then the product is $([(Me/N) \sqcup (Me'/N')], M)$: one scales the exponents.

For example $([(3, 5], 20) * ([1], 5) = ([3, 4, 5], 20)$

One only considers weight symbols such that $\prod_i (N - e_i)/e_i$ is an integer and such that the similar quotient of q-integers is a polynomial in q.

So there is a morphism to $\mathbf{N}$ and a morphism to $\mathbf{Z}[q]$.

For example $15/5 * 16/4 * 17/3 = 68$.

This is related to quasi-homogeneous isolated hypersurface singularities and their study by Milnor : pick one variable x_i of degree e_i for each exponent and choose at random a polynomial p in the x_i of total degree N. Assuming that the polynomial gives an isolated singularity, the Milnor number is given by the morphism from weight symbols to $\mathbf{N}$.

QUESTION: find an efficient algorithm to factorise a weight symbol into weight symbols.

⚠️ Factorisation is not unique, so one should rather find all possible factors. Expl $([2, 4, 6, 7], 18)$

Some examples to play with :

- type D : ([2,...,n-2],2n-2)
- Tamari posets : ([2,...,n],2n+2)
- something ([3,...,n+1],3n+3)
- Tamari intervals : ([3,...,n+1],4n+4)

Summary:

- many interesting posets around to play with
- not clear if specific and optimal algorithms are known already (Mobius, factorisation, Coxeter polynomial, etc)
- smart low-level data structure for posets ?
- derived equivalence is a difficult but rewarding problem, no public implementation of the most subtle know-how
- many open conjectures about fractionally Calabi-Yau posets, now being studied
- toy monoid of weight symbols as an investigation tool

```
In [ ]:
```