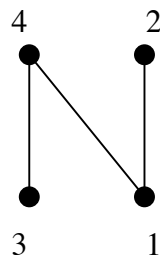


III. Data Structures

A. Representing Posets in Mathematica

In order to represent a poset P in Mathematica, we first fix a natural labeling of the poset. Suppose P has n elements. The **child form** of the poset with respect to this labeling is a list of n lists. The k -th list in the child form contains the labels for the children of the k -th element of the poset, listed in increasing order. We refer to each of these lists as **child lists**. If an element does not cover any other elements, its corresponding child list is the empty set. For example, the child form of the labeled poset shown here is $\{\{\},\{1\},\{\},\{1,3\}\}$.



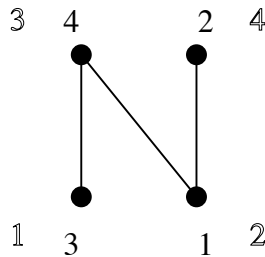
Properties of Child Forms

1. In the child form of a naturally labeled poset, the k -th child list contains only labels less than k . In particular then, the first child list is always the empty set and n does not appear in any child list.
2. The maximal elements of a poset P are exactly those elements that are not in any child list of the child form of P .

In several of our programs, we remove elements from a poset one element at a time. As we do this, the position of a child list of an element no longer corresponds with the label of the element. Thus we occasionally need to represent a poset in a different way. We do this using a list of lists where each element in the list is an ordered pair. The first member of this pair is a child list and the second member is the label of the element. We refer to this as **labeled child form**. Labeled child form is also useful when considering a convex subset of a poset. In these cases the labels for the elements are not necessarily consecutive and may not start at 1. Therefore labeling the child lists becomes necessary.

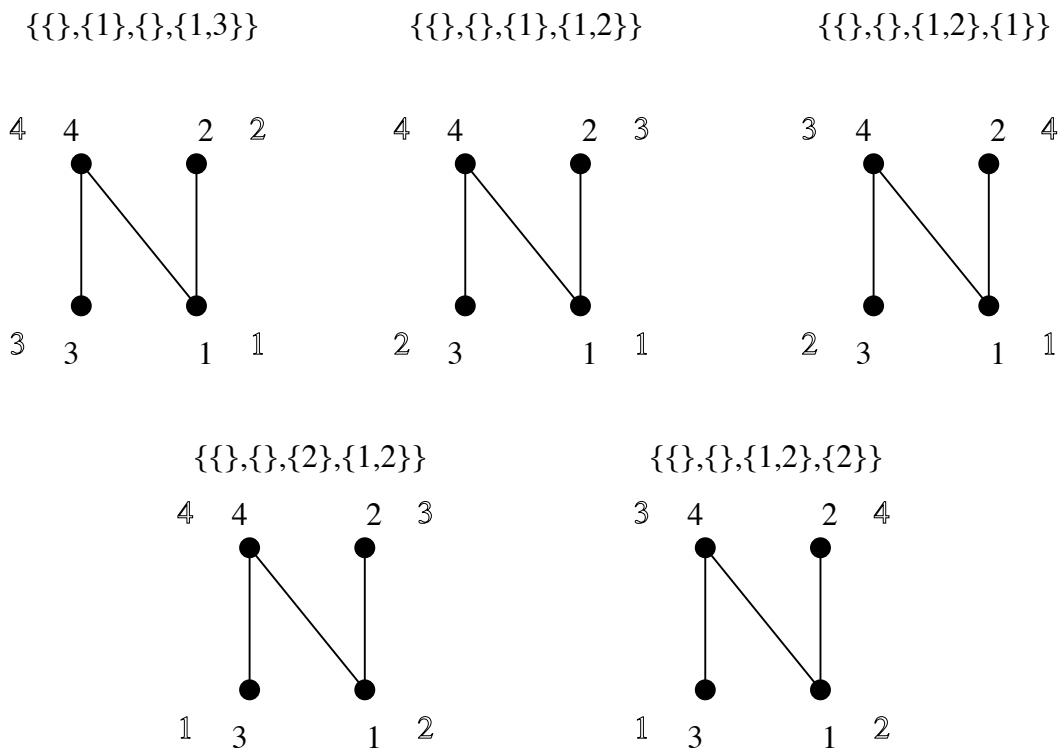
B. Standard Forms and Color Conventions

In examining posets, we often consider alternate labelings of a poset. We use different color labels to indicate how a labeling of a poset arises. We consider the initial labeling of a poset P to be black. An inverse extension provides a new labeling of the poset. A labeling of a poset obtained from an inverse extension is regarded as being yellow. To be able to obtain new labelings of a poset, we need to represent inverse extensions of posets in Mathematica. The one-rowed permutation notation allowed us to do this using a list. Given an order extension $\varphi: P \rightarrow \mathbf{n}$, the k -th element of the list of its corresponding inverse extension is $\varphi^{-1}(k)$. Consider a poset of size n with a black labeling and an inverse extension $\{\varphi^{-1}(1), \varphi^{-1}(2), \dots, \varphi^{-1}(n)\}$ consisting of black labels. We obtain a yellow labeling of the poset as follows. For each black label, we find its position in the inverse extension list. This position then becomes the new yellow label for that element of the poset. For example, the yellow labeling for the inverse extension $(3,1,4,2)$ of the poset above is:



As mentioned earlier, all the inverse extensions of P can be used to produce all alternate natural labelings of P (perhaps redundantly) and thereby we can obtain all naturally labeled posets within an isomorphism class. To prevent needless repetition in inspecting posets, we wished to identify a unique representative for each isomorphism class of posets. We call this representative the **standard form** of the posets in its isomorphism class. Mathematica provides a means to identify such a form. Suppose the child forms of all of the naturally labeled posets within a given isomorphism class are found and compiled in a list. If Mathematica is used to sort this list, we define the **standard form** for this isomorphism class to be the first child form in this sorted list (or the corresponding natural labeling). Mathematica compares posets of the same size lexicographically from the left. Suppose the first l child lists of two posets are identical. Then when the $(l+1)$ -st child lists from the two posets are compared, Mathematica first checks their lengths. It will put the child form with the shorter child list before the poset whose corresponding child list is longer. If the two corresponding child lists have the same length, Mathematica will then compare the contents of these child lists. For the child lists $\{i_1, i_2, \dots, i_k\}$ and $\{j_1, j_2, \dots, j_k\}$, let m be the first number such that $i_m \neq j_m$. Then if $i_m < j_m$, Mathematica will place the poset with the child list $\{i_1, i_2, \dots, i_k\}$ before the poset with the child list $\{j_1, j_2, \dots, j_k\}$.

To distinguish the standard labeling of a poset we consider it to be purple. Let's again consider the black labeled 4-element poset from above. The different yellow labelings and the corresponding child forms of this poset are:



If we sort the list of the child forms with Mathematica, we obtain the list $\{\{\{\},\{\},\{1\},\{1,2\}\}, \{\{\},\{\},\{1,2\},\{1\}\}, \{\{\},\{\},\{1,2\},\{2\}\}, \{\{\},\{\},\{2\},\{1,2\}\}, \{\{\},\{1\},\{\},\{1,3\}\}\}$. Then the second yellow child form shown above is the standard form for this isomorphism class. Henceforth, it is regarded as being purple.

Standard Form Proposition

1. In the standard form obtained from Mathematica, all the minimal elements are listed first. That is, the label of a minimal element will be smaller than the label of every non-minimal element.
2. Consider the following poset construction for $n > 1$. Begin with a naturally labeled poset, Q , of size $n-1$ and a new element. Label the new element n and drop an edge from this element to every maximal element of Q . This creates a naturally labeled poset P of size n with a unique maximal element labeled n . Then if Q is the standard form for its isomorphism class, P is also the standard form of its isomorphism class.

Proof

1. The child list of any minimal element is the empty set. By Mathematica's sorting rules, the empty set is smaller than any other list. Part 1 of the proposition immediately follows.
2. Let $n > 1$ and let Q and P be as described in the statement of the proposition. Suppose Q is the standard form for its isomorphism class. Let P' be a poset in the isomorphism class of P . Then P' also has a unique maximal element. By the first property of child forms, n is the maximal element. Consider the poset Q' formed by removing this maximal element from P' . Then Q' is a naturally labeled poset of size $n-1$. Since P' is isomorphic to P , there is an order-preserving bijection, φ , from P to P' such that φ^{-1} is also order-preserving. Since n is the maximal element for both P and P' , $\varphi(n) = n$. Then restricting φ to Q gives an order-preserving bijection from Q to Q' whose inverse is also order-preserving. So Q' is isomorphic to Q . Note that the child form of Q' is just the child form of P' with the last child list removed. Also the child form of P is the child form of Q with a child list containing the maximal elements of Q added to the end. In order to compare the child forms of P and P' , we first compare the child forms of Q and Q' . Since Q is the standard form for its isomorphism class, either the child form of Q' is the same as the child form of Q or there exists a k such the k -th child list of Q' is bigger (according to Mathematica's sorting rule) than the k -th child list of Q .

Case 1: Suppose the child form of Q' is not the same as the child form of Q .

So for some $1 \leq k < n$, the k -th child list of Q' is bigger than the k -th child list of Q .

Then the k -th child list of P' is bigger than the k -th child list of P .

Therefore P comes before P' in the sorted order.

Case 2: Suppose that Q' has the same child form as Q . Then the first $n-1$ child lists of P and P' agree. We now need to compare the n -th child list of these two posets. The n -th child list of P contains all maximal elements of Q . By the second property of child forms, the maximal elements of a poset are all elements not appearing in any child list. Since the child form of Q is the same as the child form of Q' , these posets must have the same maximal elements. The maximal elements of Q' are the maximal elements of $P' - \{n\}$. Since n is the only maximal element of P' , the maximal elements of Q' are the elements that appear only in the n -th child list of P' . Therefore, the n -th child list of P contains only elements that appear in the n -th child list of P' . So the n -th child list of P is either the same as the n -th child list of P' or it has fewer elements. We then have that P' has the same child form as P , or that P comes before P' in sorted order.

Either way, this is true for all posets P' in the isomorphism class of P . Therefore P is the standard form.

Along with determining standard forms, Mathematica also provides a standard way to list all of the posets of a given size: Just sort the list of the standard forms of the posets. This gives a way to uniquely name each poset of a given size: We could refer to a poset by the position of its standard form in this list. (This is not yet being done in this phase of Proctor's poset package.) The list of standard forms for $n = 3$ in standard order is $\{\{\{\},\{\},\{\}\}, \{\{\},\{\},\{1\}\}, \{\{\},\{\},\{1,2\}\}, \{\{\},\{1\},\{1\}\}, \{\{\},\{1\},\{2\}\}\}$. The first of these posets is the 3-element antichain. Observe that by Mathematica's sorting rule, the n -element antichain will always be the first poset in the list of standard forms of size n . The next poset in the standard list for $n = 3$ consists of two components: one a two-element chain, the other a single element. The next poset has the shape of an upside down V and the fourth poset is V-shape. The final poset in the list of standard forms is the 3-element chain.

We continued to use certain color conventions for our labels as we proceeded in the project. In determining whether or not a poset has the jdt or Littlewood-Richardson property, we must look at different numberings of ideals and filters. Red is used to label elements of ideals while elements of filters are labeled in green. In Chapter II, we introduced numberings and used the notation (B, Y) to represent the bi-numbering consisting of a green numbering B and a red numbering Y . In this case the order of the numberings is by color. In our programs we used a different ordering convention when representing a bi-numbering. Regardless of color, if H is a numbering of a filter of a poset P and J is a numbering of the corresponding ideal of P , this bi-numbering is given as $\{J, H\}$. In other words, the lower numbering is listed first followed by the upper numbering.

THIS PAGE IS INTENTIONALLY LEFT BLANK.