# Backtracking in Large Permutation Groups or How I Spent My Spring Vacation

Jeff Clark, Math. Dept.

September 20, 2001

I was on sabbatical during the Spring, 2001 semester. During that time I worked with Jon Berry in the Computing Sciences department on the *LINK* software package.

*LINK* was originally proposed at the Center for Discrete Mathematics and Theoretical Computer Science in July, 1991 and is currently maintained by Dr. Berry.

A large part of my work was on implementing an algorithm for determining the automorphism group of a graph. (An *automorphism* of a graph is a permutation of its vertices that sends edges to edges.)

Backtracking: using a solution tree to find one or all of the solutions for a given problem.

Problem: Find all of the elements of $S_4$ that commute with (13)(24).

Form of problem: given degree of symmetric group and a test for membership, find a way of describing the subset that satisfies this test.

Special case: the nature of the test requires that the subset be a subgroup.

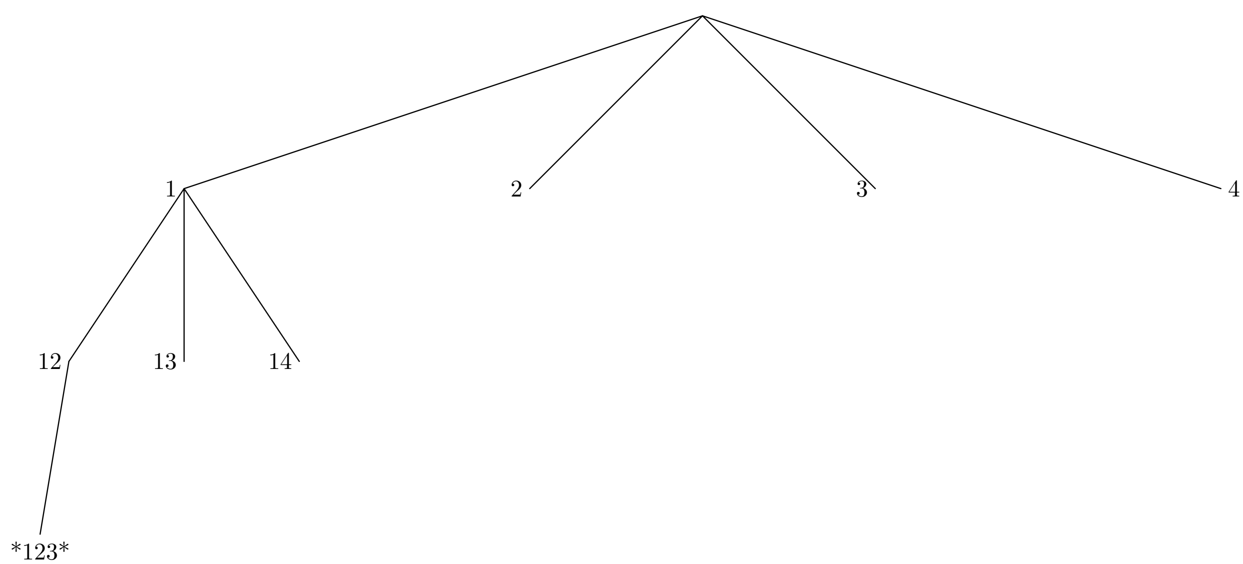
Can identify elements of $S_4$ with their images:

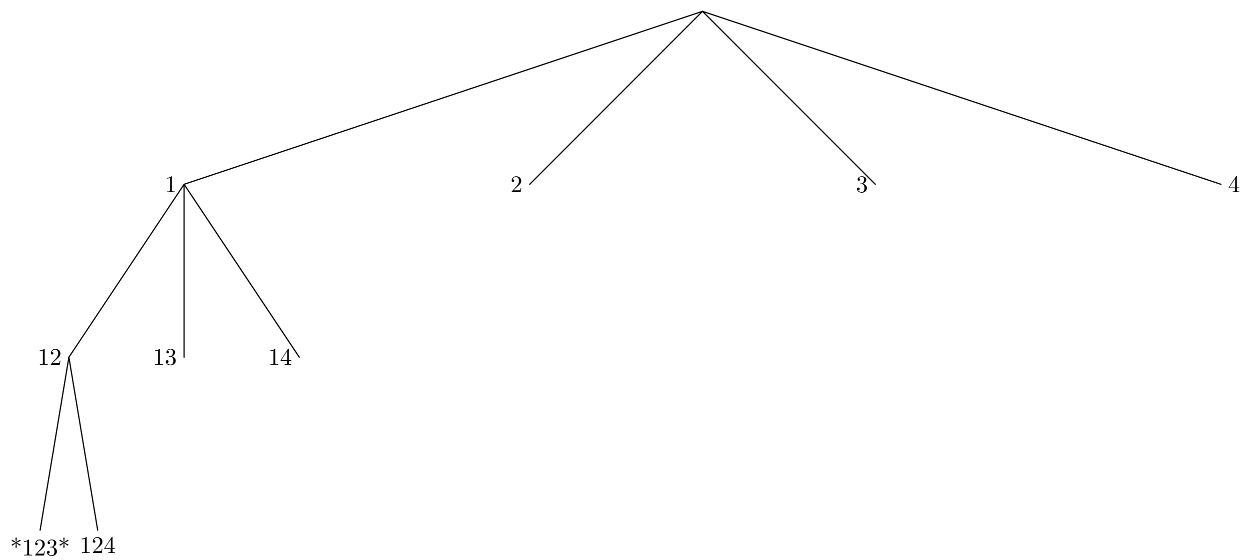
If $\sigma = (123)$ then:

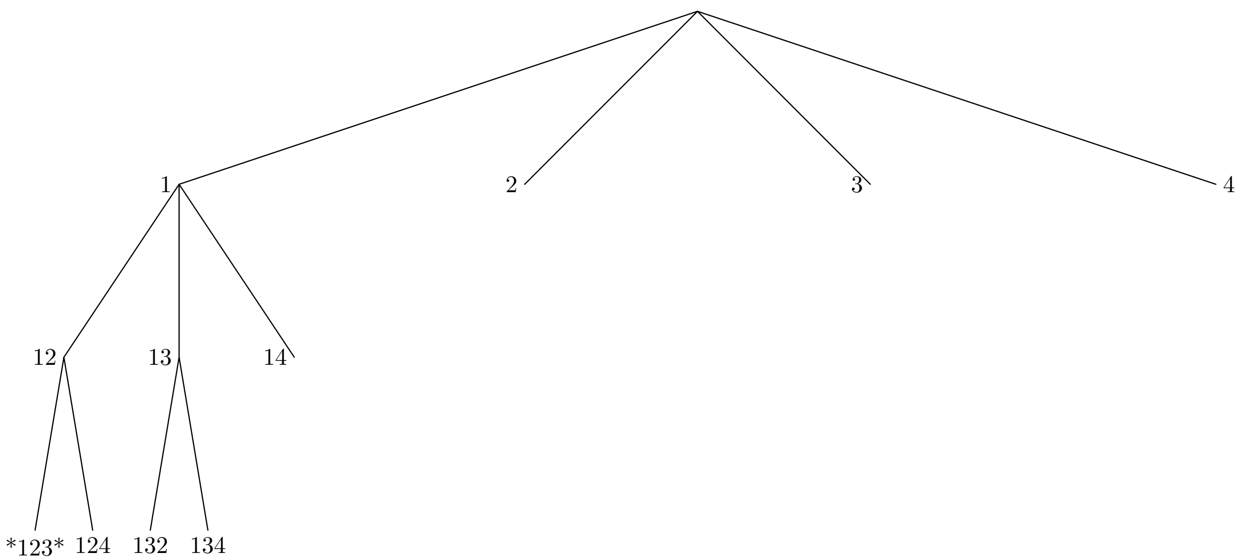
$$[\sigma(1), \sigma(2), \sigma(3), \sigma(4)] = [2, 3, 1, 4]$$

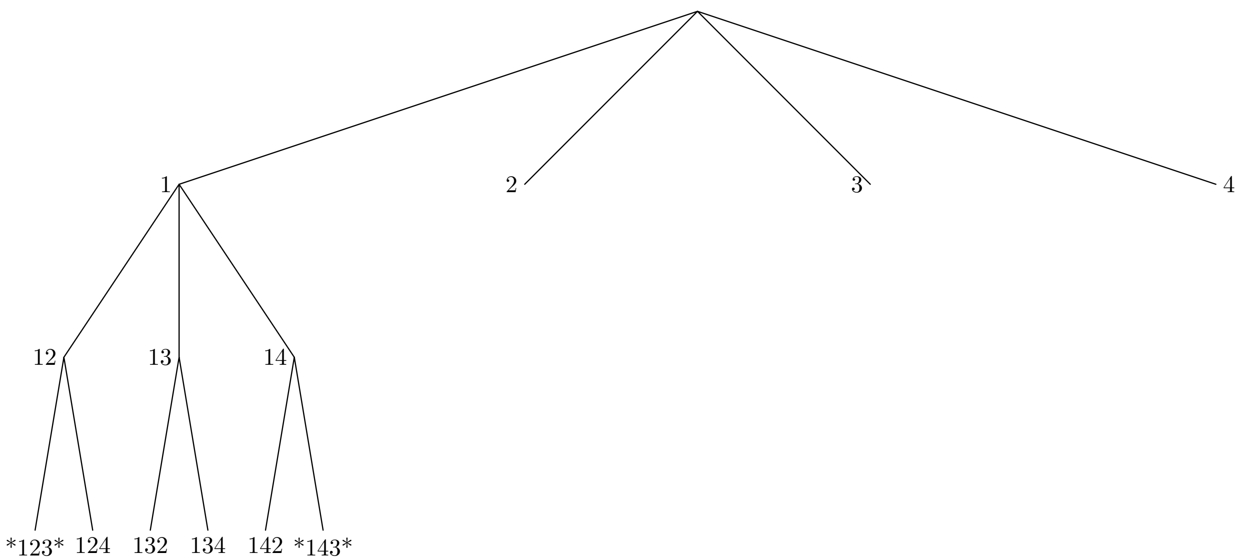
Last element is redundant; it is only element of 1–4 that is left.

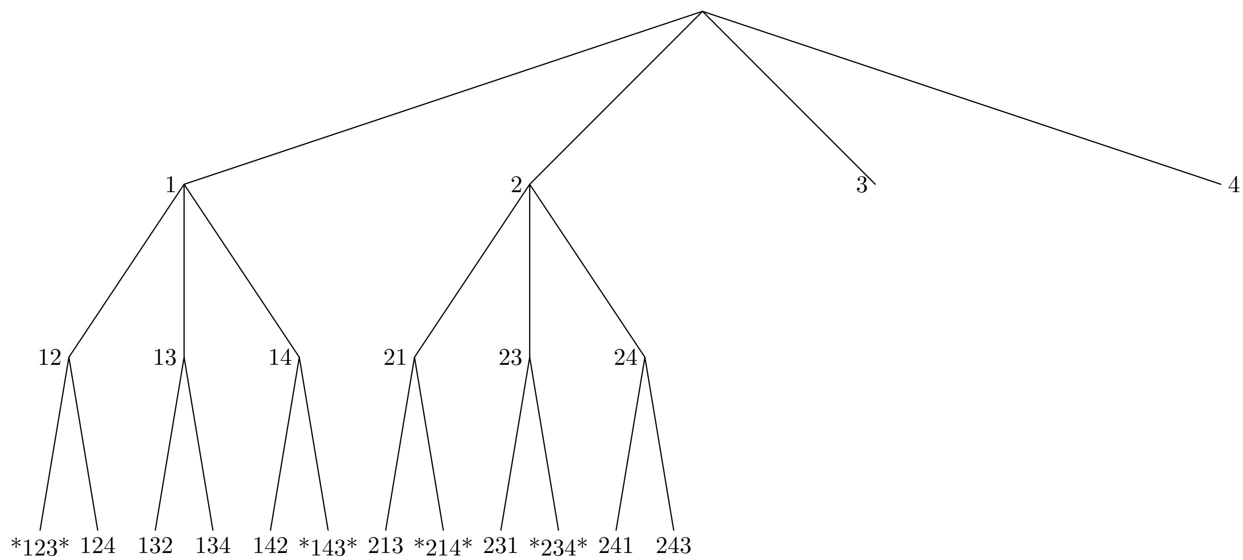
Can identify elements of $S_4$ with their image on the first three points 1–3.


1
2
3
4
12
13
14
*123*


1
2
3
4
12
13
14
*123*
124


1
2
3
4
12
13
14
*123*
124
132
134


1
2
3
4
12
13
14
*123*
124
132
134
142
*143*


1
2
3
4
12
13
14
21
23
24
*123*
124
132
134
142
*143*
213
*214*
231
*234*
241
243

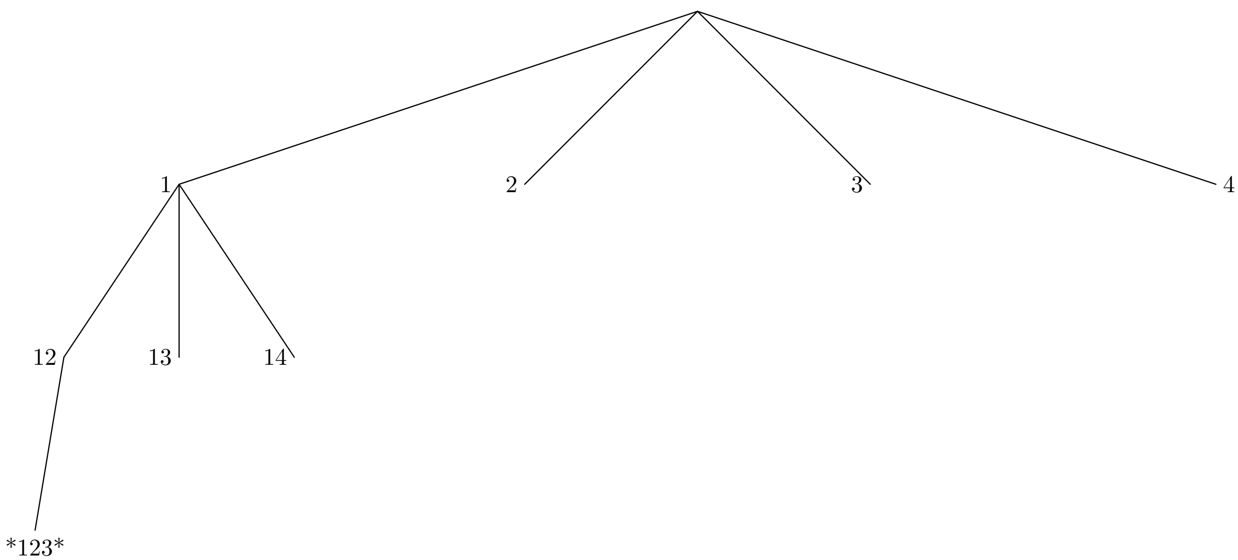
This process ignores the structure of the solution set: it is a subgroup of $S_4$.

Rather than enumerate the solution set (which will be lengthy), it would be more efficient to find a set of generators for the solution set.

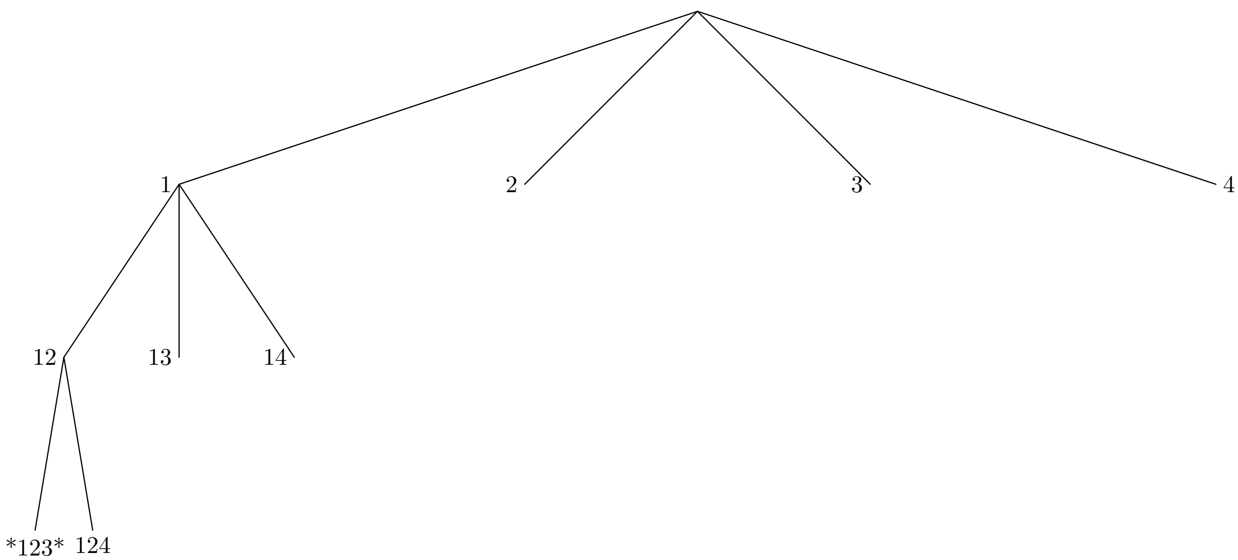
Let $G$ be the solution set. Let $G_i$ be the subgroup of $G$ that fixes the elements 1 through $i$.

The procedure will be recursive: at stage $k$ we will have a generating set for $G_k$, and we will seek to move to stage $k-1$ until $k=0$.

We begin at the node at the bottom left of the decision tree: 123 which is the identity permutation. It is our generator for $G_3$. (Since it is a trivial generator, it can be omitted from the final result.)

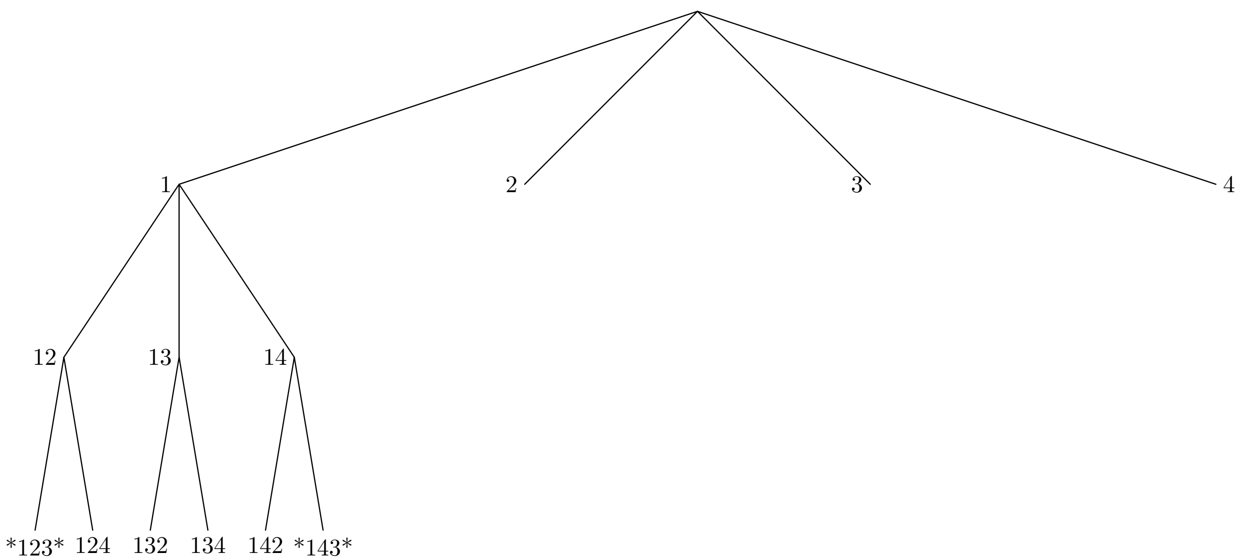

1
2
3
4
12
13
14
*123*

We go up to the next level and examine the elements of our solution tree that fix the first $k = 2$ elements:


1
2
3
4
12
13
14
*123*
124

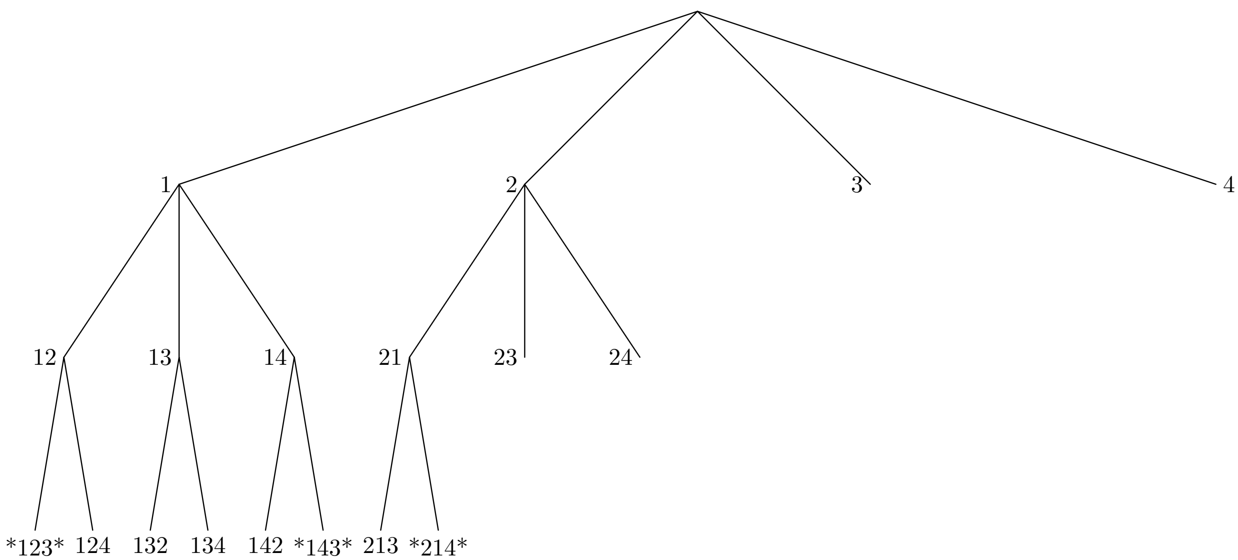
The only element to check is 124, i.e., (34) which is not a solution.

We go up to the next level and examine the elements of our solution tree that fix the first $k = 1$ elements:


1
2
3
4
12
13
14
*123*
124
132
134
142
*143*

We stop for the time being as soon as we find a solution: 143, i.e., (24). Since this is the last node to check for $G_1$, we have that $G_1$ is generated by (24).

At this step in the process, the orbits of the elements $\{1, 2, 3, 4\}$ under $G_1$ are $\{1\}$, $\{2, 4\}$, and $\{3\}$. As I move to checking $G_0$, I will have at most to look at the trees headed by 2 and 3. For if $\tau$ is a permutation that sends 1 to 4, then $\tau(24)^{-1}$ is a permutation that sends 1 to 2, and $\tau$ can be expressed in terms of any generators coming from the tree headed by 2 and the generators of $G_1$.


1
2
3
4
12
13
14
21
23
24
*123*
124
132
134
142
*143*
213
*214*

We find that 214, i.e., (12)(34) is a generator for $G_0$. With it and (24), the only orbit of the elements $\{1, 2, 3, 4\}$ is $\{1, 2, 3, 4\}$ itself. By the same argument as before, I don't need to look for any other solutions $\tau$, because if $\tau(1) = k$, there is an element $\nu$ expressible in terms of our generators (12)(34) and (24) that sends 1 to $k$, and $\tau\nu^{-1}$ will fix 1, be contained in $G_1$ and thus be expressible in terms of the generators already found.

For this problem, $G$ is generated by (12)(34) and (24). Rather than enumerate all 24 possible solutions, we only had to check 8 possible solutions.

Algorithm:

1. Let $n$ be the degree of the group. Start with $G_{n-1} = <>$. Let $k = n - 1$. Let $S = \{\}$ be the set of generators found so far. Let $O = \{\{1\}, \{2\}, \{3\}, \{4\}\}$ be the orbits of $\{1, 2, 3, 4\}$ under the generators in $S$.

2. If $k = 0$, stop.

3. Set $k := k - 1$.

4. Enumerate the elements of $G_k$ that send $k+1$ to an element that is the smallest possible element in its orbit until one is found that satisfies the test for membership.

5. If no such element, return to Step #3.

6. If $\tau$ is such an element, add it to $S$. Recompute $O$ to see if any of the orbits have merged. Return to Step #4.

The generating set $S$ is a *strong generating set*: $G_k =< G_k \cap S >$. This means that we can find the size of $G$, by taking the product of $[G_{k-1} : G_k]$, which is the orbit of the point $k$ under $G_{k-1}$.

The elements of the orbit of $k$ under $G_{k-1}$ can also be used to enumerate $G$.

In *LINK*, the degree of the permutation group is the number of vertices, and the test for membership in the solution set is a test as to whether or not the permutation is an automorphism.