群論

Maple 17 では、群 (とくに順列の群として表現される有限群) を扱うための新しいパッケージが導入されました。

```
> restart :
```

```
> with(GroupTheory);
```

```
[ <|>, AllSmallGroups, AllTransitiveGroups, Alt, AlternatingGroup, AreConjugate,                (1)
  AreIsomorphic, BabyMonster, CayleyTable, CayleyTableGroup, Center, Centraliser,
  Centralizer, Centre, ConjugacyClass, ConjugacyClasses, Conjugator, ConwayGroup,
  Core, CustomGroup, CyclicGroup, Degree, DerivedLength, DerivedSeries,
  DerivedSubgroup, DicyclicGroup, DihedralGroup, DirectProduct,
  DirectProductImplementation, DrawCayleyTable, DrawSubgroupLattice,
  ElementaryGroup, Elements, Embedding, ExceptionalGroup, FPGroup, Factor,
  FischerGroup, FittingSubgroup, FrattiniSubgroup, GL, GaloisGroup,
  GeneralLinearGroup, GeneralOrthogonalGroup, GeneralUnitaryGroup, Generators,
  Group, GroupOrder, HaradaNortonGroup, HeldGroup, HigmanSimsGroup,
  Hypercentre, IdentifySmallGroup, Index, Intersection, IsAbelian, IsAlternating,
  IsCommutative, IsElementary, IsFinite, IsNilpotent, IsNormal, IsPerfect, IsPrimitive,
  IsRegular, IsSimple, IsSoluble, IsSolvable, IsSubgroup, IsSymmetric, IsTransitive,
  JankoGroup, Labels, LeftCoset, LeftCosets, LowerCentralSeries, LyonsGroup,
  MathieuGroup, McLaughlinGroup, MetacyclicGroup, Monster, NilpotencyClass,
  NilpotentResidual, NonRedundantGenerators, NormalClosure, Normaliser,
  NormaliserSubgroup, NormalizerSubgroup, NumGroups, NumTransitiveGroups,
  ONanGroup, Operations, Orbit, Orbits, OrthogonalGroup, PCore, PGL, PGU, PSL,
  PSU, PSp, PermApply, PermCommutator, PermConjugate, PermCycleType, PermDegree,
  PermFixed, PermInverse, PermLeftQuotient, PermOrder, PermParity, PermPower,
  PermProduct, PermRightQuotient, PermSupport, PermutationGroup,
  ProjectiveGeneralLinearGroup, ProjectiveGeneralUnitaryGroup,
  ProjectiveSpecialLinearGroup, ProjectiveSpecialUnitaryGroup,
  ProjectiveSymplecticGroup, QuaternionGroup, RandomElement, Relators, RightCoset,
  RightCosets, RubiksCubeGroup, RudvalisGroup, SL, SmallGroup, SolubleResidual,
  SolvableResidual, SpecialLinearGroup, SpecialOrthogonalGroup, SpecialUnitaryGroup,
  Stabiliser, Stabilizer, Subgroup, SubgroupLattice, SubgroupMembership, Supergroup,
  SuzukiGroup, SylowSubgroup, Symm, SymmetricGroup, SymplecticGroup,
  ThompsonGroup, TitsGroup, TransitiveGroup, TrivialGroup, UpperCentralSeries ]
```

順列

順列は、Perm コンストラクタを使用して作成されます。順列を使用する演算は、非可換乗算 (.) の影響を受けます。また、順列を使用する累乗 (^) は共役または指数として認識されます。

```
> a := Perm( [ [1, 2] ]);
```

$$a := (1, 2) \tag{1.1}$$

> $b := Perm([[1, 3, 5], [2, 4]])$

$$b := (1, 3, 5)(2, 4) \tag{1.2}$$

> $PermProduct(a, b) = a.b$

$$(1, 4, 2, 3, 5) = (1, 4, 2, 3, 5) \tag{1.3}$$

> $PermConjugate(a, b) = a^b$

$$(3, 4) = (3, 4) \tag{1.4}$$

> $PermInverse(a) = a^{-1}$

$$(1, 2) = (1, 2) \tag{1.5}$$

> $PermCommutator(a, b) = a^{-1}.b^{-1}.a.b$

$$(1, 2)(3, 4) = (1, 2)(3, 4) \tag{1.6}$$

順列群

このリリースでは、順列群 (複数の正の整数 n に対する集合 {1,2,...,n} となる有限集合における順列の有限集合によって生成される群) に重点的に取り組みました。任意の (有限) 群に適用可能な抽象群のプロパティおよび計算に加えて、順列群に固有の計算も選択できるようになりました。

> $G := Group(Perm([[1, 2], [3, 4]]), Perm([[1, 2, 3, 4]]))$

$$G := \langle (1, 2)(3, 4), (1, 2, 3, 4) \rangle \tag{2.1}$$

> $GroupOrder(G)$

$$8 \tag{2.2}$$

> $IsTransitive(G)$

$$\text{true} \tag{2.3}$$

> $IsPrimitive(G)$

$$\text{false} \tag{2.4}$$

> $IsRegular(G)$

$$\text{false} \tag{2.5}$$

> $IsNilpotent(G)$

$$\text{true} \tag{2.6}$$

> $G := PSL(3, 3)$

$$G := \mathbf{PSL}(3, 3) \tag{2.7}$$

> $GroupOrder(G)$

$$5616 \tag{2.8}$$

> $Generators(G)$

$$[(5, 8, 11)(6, 9, 12)(7, 10, 13), (1, 2, 5)(3, 8, 7)(4, 11, 6)(9, 10, 13)] \tag{2.9}$$

> $IsPrimitive(G)$

$$\text{true} \tag{2.10}$$

> $G := RubiksCubeGroup( ) :$

> *IsTransitive*(*G*)

false (2.11)

> *O1* := *Orbit*(1, *G*)

O1 := (2.12)

$$1^{\langle (6,25,43,16)(7,28,42,13)(8,30,41,11)(17,19,24,22)(18,21,23,20),\ (1,14,48,27)(2,12,47,29)(3,9,46,32)(33,35,40,38)(34,37,39,36),\ (1,17,41,40)(4,20,44,37)(6,22,46,35)(9,11,16,14)(10,13,15,12),\ (3,38,43,19)(5,36,45,21)(8,33,48,24)(25,27,32,30)(26,29,31,28),\ (1,3,8,6)(2,5,7,4)(9,33,25,17)(10,34,26,18)(11,35,27,19),\ (14,22,30,38)(15,23,31,39)(16,24,32,40)(41,43,48,46)(42,45,47,44)\rangle}$$

> *Elements*(*O1*)

{1, 3, 6, 8, 9, 11, 14, 16, 17, 19, 22, 24, 25, 27, 30, 32, 33, 35, 38, 40, 41, 43, 46, 48} (2.13)

CayleyTable 群

Cayley テーブル (群の演算テーブル) で群を定義することができます。Cayley テーブルは整数の配列または行列として指定されます。

> $ct := \langle\langle 1|2|3\rangle, \langle 2|3|1\rangle, \langle 3|1|2\rangle\rangle$

$$ct := \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{bmatrix} \quad (3.1)$$

> *G* := *Group*(*ct*)

G := < *a Cayley table group with 3 elements* > (3.2)

> *IsAbelian*(*G*)

true (3.3)

有限表現群

一連のジェネレータを指定して関係を定義することで、群を定義できます。

たとえば、これは 5 度の交代群の定義です。

> $G := \langle x, y | \langle x^2 = 1, y^3 = 1, (x.y)^5 = 1\rangle\rangle$

$$G := \langle x, y \,|\, x^2, y^3, x\,y\,x\,y\,x\,y\,x\,y\,x\,y\rangle \quad (4.1)$$

群の次数を直接検証することができます。

> *GroupOrder*(*G*)

60 (4.2)

簡素さをチェックするために、まず有限的に表現された群を順列群に変換します。次に、IsSimple コマンドを使用します。

> *IsSimple*(*PermutationGroup*(*G*))

true (4.3)

記号群

記号群は、ゼロ個以上の「パラメータ」(通常は整数) に依存する群を表すために使用されます。たとえば、群の要素を用いた実際的計算で対応するには大規模すぎる散在型の有限単純群 (モンスターおよび原田-ノートン単純群など) を表すために使用されます。さらに、一部のコンストラクタは、パラメータに記号式を使用して、構築された群の記号表現を返すことができます。

> *Monster*( )

$$\mathbb{M} \tag{5.1}$$

> *GroupOrder*(*Monster*( ))

$$808017424794512875886459904961710757005754368000000000 \tag{5.2}$$

> *IsSimple*(*Monster*( ))

$$\text{true} \tag{5.3}$$

> $G := GeneralLinearGroup(3, q)$

$$G := \mathbf{GL}(3, q) \tag{5.4}$$

> *GroupOrder*(*G*)

$$(q^3 - 1)\,(q^3 - q)\,(q^3 - q^2) \tag{5.5}$$

> *IsSimple*(*Alt*(*n*)) assuming $5 < n$

$$\text{true} \tag{5.6}$$

群コンストラクタ

GroupTheory パッケージで用意されている専用のコンストラクタを使用して、多くの群を直接かつ便利に作成できます。

> *Symm*(4)

$$\mathbf{S}_4 \tag{6.1}$$

> *Alt*(6)

$$\mathbf{A}_6 \tag{6.2}$$

> *DihedralGroup*(14)

$$\mathbf{D}_{14} \tag{6.3}$$

> *PSL*(3, 4)

$$\mathbf{PSL}(3, 4) \tag{6.4}$$

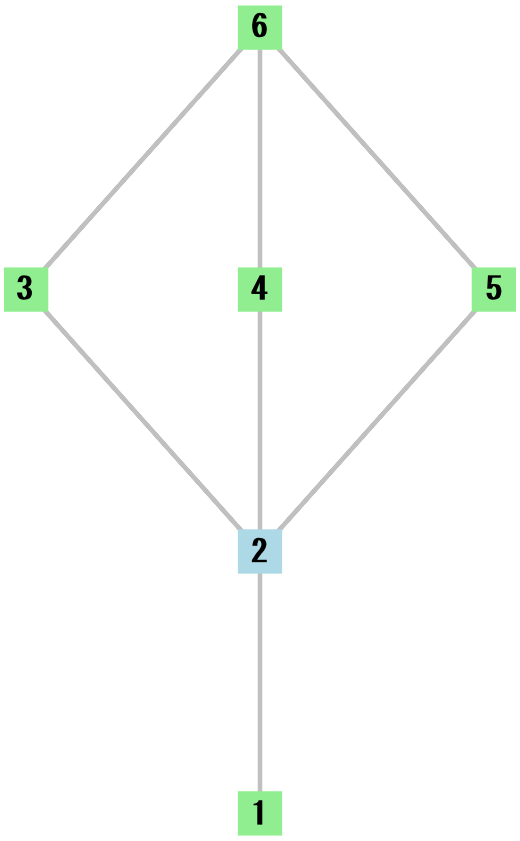
> *GL*(2, 2)

$$\mathbf{GL}(2, 2) \tag{6.5}$$

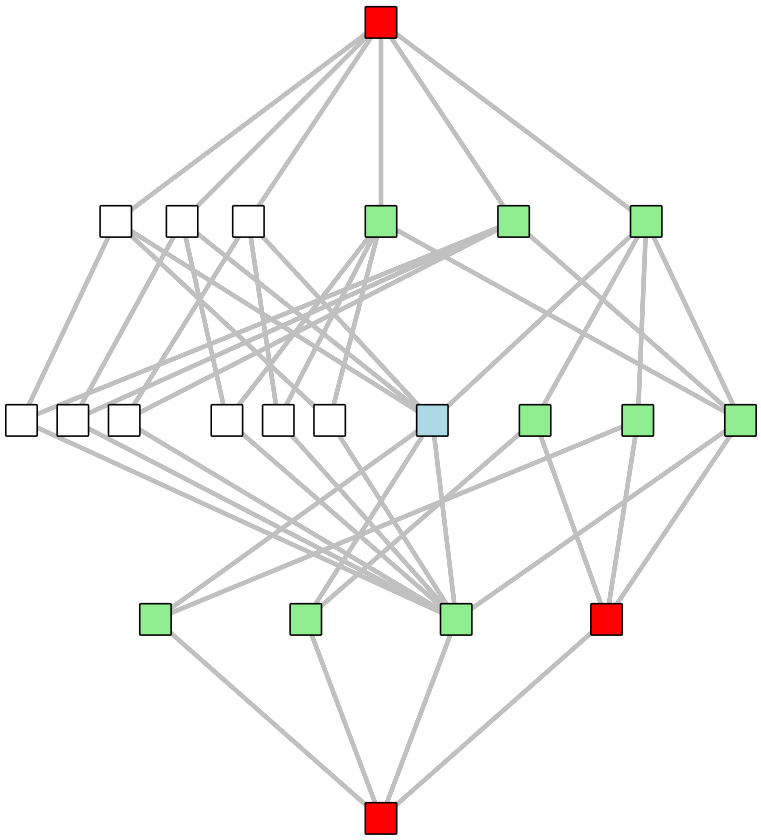
可視化

部分群の格子をグラフィカルに調べて群を可視化することができます。

> *DrawSubgroupLattice*(*QuaternionGroup*( ))




```
> G := SmallGroup(24, 7) :
> DrawSubgroupLattice(G, derived, labels = none)
```



> *DrawCayleyTable*(*Symm*(3))

	( )	(1, 2)	(1, 2, 3)	(1, 3)	(2, 3)	(1, 3, 2)
( )	( )	(1, 2)	(1, 2, 3)	(1, 3)	(2, 3)	(1, 3, 2)
(1, 2)	(1, 2)	( )	(1, 3)	(1, 2, 3)	(1, 3, 2)	(2, 3)
(1, 2, 3)	(1, 2, 3)	(2, 3)	(1, 3, 2)	(1, 2)	(1, 3)	( )
(1, 3)	(1, 3)	(1, 3, 2)	(2, 3)	( )	(1, 2, 3)	(1, 2)
(2, 3)	(2, 3)	(1, 2, 3)	(1, 2)	(1, 3, 2)	( )	(1, 3)
(1, 3, 2)	(1, 3, 2)	(1, 3)	( )	(2, 3)	(1, 2)	(1, 2, 3)

群プロパティ

GroupTheory パッケージでは、群のさまざまなプロパティをテストすることができます。

> $G := Symm(3)$

$$G := \mathbf{S}_3 \tag{8.1}$$

> $IsAbelian(G)$

$$\text{false} \tag{8.2}$$

> $IsNilpotent(G)$

$$\text{false} \tag{8.3}$$

> $IsSoluble(G)$

$$\text{true} \tag{8.4}$$

> $IsSimple(G)$

$$\text{false} \tag{8.5}$$

さまざまな部分群

GroupTheory パッケージを使用して、多くの標準的な部分群を構築できます。

> $G := DihedralGroup(8)$

$$G := D_8 \quad (9.1)$$

```
> Z := Centre(G)
```

$$Z := Z(D_8) \quad (9.2)$$

```
> GroupOrder(Z)
```

$$2 \quad (9.3)$$

```
> C := DerivedSubgroup(G)
```

$$C := [D_8, D_8] \quad (9.4)$$

```
> GroupOrder(C)
```

$$4 \quad (9.5)$$

```
> P := PCore(2, MetacyclicGroup(3, 2, 2))
```

$$P := O_2(\langle (1,3,4)(2,5,6), (1,2)(3,5)(4,6) \rangle) \quad (9.6)$$

```
> LowerCentralSeries(P)
```

$$O_2(\langle (1,3,4)(2,5,6), (1,2)(3,5)(4,6) \rangle) \rhd \langle \rangle \quad (9.7)$$

```
> R := SolubleResidual(Symm(6))
```

$$R := \langle (1,3,2), (2,4,3), (3,5,4), (4,6,5) \rangle \quad (9.8)$$

```
> IsPerfect(R)
```

$$\text{true} \quad (9.9)$$

同型テストおよび群の同定

GroupTheory パッケージには、それぞれの Cayley テーブルで与えられた有限群に対する同型テストが含まれています。Cayley テーブルを計算することで、このコマンドの演算が他の有限群に拡張されます (これは自動的に行われます)。また、パッケージには、次数が 200 未満のすべての有限群のデータベースが含まれます。　これにより、標準的な方法で最大 200 個の要素を持つ有限群を一義的に同定できます。

AreIsomorphic コマンドを使用することで、同型までの直積構築の可換性を検証できます。

```
> G := DirectProduct(Alt(4), DihedralGroup(6))
```

$$G := A_4 \times D_6 \quad (10.1)$$

```
> H := DirectProduct(DihedralGroup(6), Alt(4))
```

$$H := D_6 \times A_4 \quad (10.2)$$

```
> AreIsomorphic(G, H)
```

$$\text{true} \quad (10.3)$$

```
> IdentifySmallGroup(G)
```

$$144, 190 \quad (10.4)$$

参照

GroupTheory パッケージ