SmallClassNr

Library of finite groups with small class number

1.5.0

5 April 2026

Sam Tertooy

Sam Tertooy Email: sam.tertooy@kuleuven.be
Homepage: https://stertooy.github.io/
Address: Wiskunde
KU Leuven, Kulak Kortrijk Campus
Etienne Sabbelaan 53
8500 Kortrijk
Belgium

Abstract

The SmallClassNr package provides access to finite groups with small class number. Currently, it contains the finite groups of class number at most 14.

Copyright

Acknowledgements

This documentation was created using the GAPDoc and AutoDoc packages.

Contents

Chapter 1

The SmallClassNr package

This is the manual for the GAP 4 package SmallClassNr version 1.5.0, developed by Sam Tertooy.

1.1 Installation

If you are using GAP version 4.15.0 or newer, then SmallClassNr should be installed by default.

If this is not the case, but the PackageManager package is installed and loaded, you can install SmallClassNr from within a GAP session using `InstallPackage` (**PackageManager: InstallPackage**).

```
gap> InstallPackage( "SmallClassNr" );
...
true
```

Alternatively, you can download SmallClassNr as a .tar.gz archive here. After extracting, you should place it in a suitable `pkg` folder. For example, on a Debian-based Linux distribution (e.g. Ubuntu, Mint), you can place it in `$HOME/.gap/pkg` (recommended) which makes it available for just yourself, or in the GAP installation directory (`gap-X.Y.Z/pkg`) which makes it available for all users.

You can use the following command to efficiently install the package for yourself:

```
wget -qO - https://[...].tar.gz | tar xzf - --one-top-level=$HOME/.gap/pkg
```

1.2 Loading

Once installed, loading SmallClassNr can be done by using `LoadPackage` (**Reference: LoadPackage**).

```
gap> LoadPackage( "SmallClassNr" );
...
true
```

1.3 Citing

If you use the SmallClassNr package in your research, we would love to hear about your work via an email to the address sam.tertooy@kuleuven.be. If you have used the SmallClassNr package in the preparation of a paper and wish to refer to it, please cite it as described below.

In BibTeX:

BibTeX

```
@misc{SCN1.5.0,
    author =       {Tertooy, Sam},
    title =        {{SmallClassNr,
                     Library of finite groups with small class number,
                     Version 1.5.0}},
    note =         {GAP package},
    year =         {2026},
    howpublished = {\url{https://stertooy.github.io/SmallClassNr}}
}
```

In BibLaTeX:

BibLaTeX

```
@software{SCN1.5.0,
    author =   {Tertooy, Sam},
    title =    {SmallClassNr},
    subtitle = {Library of finite groups with small class number},
    version =  {1.5.0},
    note =     {GAP package},
    year =     {2026},
    url =      {https://stertooy.github.io/SmallClassNr}
}
```

1.4 Support

If you encounter any problems, please submit them to the issue tracker. If you have any questions on the usage or functionality of SmallClassNr, you may contact me via email at sam.tertooy@kuleuven.be.

Chapter 2

Classification

The *class number* $k(G)$ of a group G is the number of conjugacy classes of G. In 1903, Landau proved in [**?**] that for every $n \in \mathbb{N}$, there are only finitely many finite groups with exactly n conjugacy classes. The SmallClassNr package provides access to the finite groups with class number at most 14. These groups were classified in the following papers:

- $k(G) \leq 5$, by Miller in [**?**] and independently by Burnside in [**?**]
- $k(G) = 6, 7$, by Poland in [**?**]
- $k(G) = 8$, by Kosvintsev in [**?**]
- $k(G) = 9$, by Odincov and Starostin in [**?**]
- $k(G) = 10, 11$, by Vera López and Vera López in [**?**]
- $k(G) = 12$, by Vera López and Vera López in [**?**]
- $k(G) = 13, 14$, by Vera López and Sangroniz in [**?**]

Remarks:

1. In [**?**], three distinct groups of the form $(C_5 \times C_5) \rtimes C_4$ order 100 with class number 10 are given. However, only two such groups exist, being the ones with `IdClassNr` equal to `[10,25]` and `[10,26]`.

2. In [**?**], 48 groups with class number 12 are listed. There are actually 51 such groups, the three groups missing in [**?**] are provided in the appendix of [**?**]. These are the groups with `IdClassNr` equal to `[12,13]`, `[12,16]` and `[12,39]`.

Chapter 3

The Small Class Number Library

3.1 Selecting groups by their ID's

3.1.1 SmallClassNrGroup

▷ SmallClassNrGroup(*k*, *i*) (function)
▷ SmallClassNrGroup(*k*, *i*: *AsPermGroup*) (function)

Returns: the *i*-th finite group of class number *k* in the library.

By default, if the group is soluble, it is given as a PcGroup whose Pcgs is a SpecialPcgs. If the group is not soluble, or if the option `AsPermGroup` is added, it will be given as a permutation group of minimal permutation degree and with a minimal generating set.

Example
```
gap> SmallClassNrGroup( 4, 4 );
<pc group of size 12 with 3 generators>
gap> G := SmallClassNrGroup( 4, 4 : AsPermGroup );
Group([ (1,2,3), (1,4,2) ])
gap> NrConjugacyClasses( G );
4
gap> IsAlternatingGroup( G );
true
```

3.1.2 IdClassNr

▷ IdClassNr(*G*) (attribute)

Returns: the SmallClassNr ID of *G*, i.e. a pair [*k*, *i*] such that *G* is isomorphic to `SmallClassNrGroup(`*k*, *i*`)`.

Example
```
gap> IdClassNr( AlternatingGroup( 4 ) );
[ 4, 4 ]
```

3.2 Selecting groups by their properties

For each of the functions in this section, the arguments *arg* must come in pairs consisting of a function and a value (or list of accepted values). At least one of the functions must be `NrConjugacyClasses`. Missing functions will be interpreted as `NrConjugacyClasses`, missing values as `true`.

The option `AsPermGroup` can be added to the functions in this section to ensure that all groups are returned as PermGroups (instead of PcGroups if they are soluble).

3.2.1 AllSmallClassNrGroups

▷ `AllSmallClassNrGroups(arg...)` (function)
▷ `AllSmallClassNrGroups(arg...: AsPermGroup)` (function)

Returns: all finite groups with certain properties as specified by `arg`.

Example
```
gap> AllSmallClassNrGroups( IsSolvable, true, NrConjugacyClasses, 6 );
[ <pc group of size 6 with 2 generators>,
  <pc group of size 12 with 3 generators>,
  <pc group of size 12 with 3 generators>,
  <pc group of size 18 with 3 generators>,
  <pc group of size 18 with 3 generators>,
  <pc group of size 36 with 4 generators>,
  <pc group of size 72 with 5 generators> ]
gap> AllSmallClassNrGroups( [ 3 .. 5 ], IsNilpotent );
[ <pc group of size 3 with 1 generator>,
  <pc group of size 4 with 2 generators>,
  <pc group of size 4 with 2 generators>,
  <pc group of size 5 with 1 generator>,
  <pc group of size 8 with 3 generators>,
  <pc group of size 8 with 3 generators> ]
gap> AllSmallClassNrGroups( [ 3 .. 5 ], IsNilpotent : AsPermGroup );
[ Group([ (1,2,3) ]),
  Group([ (1,2,3,4) ]),
  Group([ (1,2), (3,4) ]),
  Group([ (1,2,3,4,5) ]),
  Group([ (1,2), (1,3)(2,4) ]),
  Group([ (1,2,3,4)(5,6,7,8), (1,5,3,7)(2,8,4,6) ]) ]
```

3.2.2 OneSmallClassNrGroup

▷ `OneSmallClassNrGroup(arg...)` (function)
▷ `OneSmallClassNrGroup(arg...: AsPermGroup)` (function)

Returns: one finite group with certain properties as specified by `arg`.

Example
```
gap> OneSmallClassNrGroup( 6, IsSolvable, false );
Group([ (1,2,3)(4,5,6), (1,4)(2,7) ])
gap> OneSmallClassNrGroup( 10, IsSolvable, true, IsNilpotent, false );
<pc group of size 28 with 3 generators>
gap> OneSmallClassNrGroup( 10, IsSolvable, true, IsNilpotent, false : AsPermGroup );
Group([ (1,2,3,4,5,6,7), (2,7)(3,6)(4,5)(8,9,10,11) ])
```

3.2.3 NrSmallClassNrGroups

▷ `NrSmallClassNrGroups(arg...)` (function)

Returns: the number of finite groups with certain properties as specified by `arg`.

Example

```
gap> NrSmallClassNrGroups( 14 );
93
gap> NrSmallClassNrGroups( IsSolvable, true, NrConjugacyClasses, 6 );
7
gap> NrSmallClassNrGroups( [ 3 .. 5 ], IsNilpotentGroup );
6
```

3.2.4 IteratorSmallClassNrGroups

▷ `IteratorSmallClassNrGroups(`*arg...*`)` (function)

Returns: an iterator that iterates over the finite groups with properties as specified by *arg*.

Example

```
gap> iter := IteratorSmallClassNrGroups( 12, IsSimpleGroup );
<iterator>
gap> for G in iter do Print( Size( G ), "\n" ); od;
3420
5616
443520
```

3.3 Availability of the library

3.3.1 SmallClassNrGroupsAvailable

▷ `SmallClassNrGroupsAvailable(`*k*`)` (function)

Returns: `true` if the finite groups of class number *k* are available in the library, and `false` otherwise.

Example

```
gap> SmallClassNrGroupsAvailable( 14 );
true
gap> SmallClassNrGroupsAvailable( 15 );
false
```

Chapter 4

Conversion to other group libraries

4.1 The Small Groups Library

This library is provided by the SmallGrp package.

4.1.1 IdClassNrToIdGroup

▷ `IdClassNrToIdGroup(k, i)` (function)

Returns: a pair of integers `[x, y]` such that `SmallGroup(x, y)` is isomorphic to `SmallClassNrGroup(k, i)`.

Example
```
gap> IdClassNrToIdGroup( 9, 19 );
[ 192, 1025 ]
gap> IdClassNr( SmallGroup( 192, 1025 ) );
[ 9, 19 ]
```

4.2 The Library of Finite Perfect Groups

This library is provided by GAP itself.

4.2.1 IdClassNrToPerfGrp

▷ `IdClassNrToPerfGrp(k, i)` (function)

Returns: a pair of integers `[x, y]` such that `PerfectGroup(x, y)` is isomorphic to `SmallClassNrGroup(k, i)`.

Example
```
gap> IdClassNrToPerfGrp( 10, 36 );
[ 14520, 1 ]
gap> IdClassNr( PerfectGroup( 14520, 1 ) );
[ 10, 36 ]
```

4.3 The Primitive Permutation Groups Library

This library is provided by the PrimGrp package.

4.3.1 IdClassNrToPrimGrp

▷ IdClassNrToPrimGrp(*k*, *i*) (function)

Returns: a pair of integers [x, y] such that PrimitiveGroup(x, y) is isomorphic to SmallClassNrGroup(*k*, *i*).

Example

```
gap> IdClassNrToPrimGrp( 9, 25 );
[ 49, 25 ]
gap> IdClassNr( PrimitiveGroup( 49, 25 ) );
[ 9, 25 ]
```

4.4 The Library of Transitive Groups

This library is provided by the TransGrp package.

4.4.1 IdClassNrToTransGrp

▷ IdClassNrToTransGrp(*k*, *i*) (function)

Returns: a pair of integers [x, y] such that TransitiveGroup(x, y) is isomorphic to SmallClassNrGroup(*k*, *i*).

Example

```
gap> IdClassNrToTransGrp( 12, 46 );
[ 45, 314 ]
gap> IdClassNr( TransitiveGroup( 45, 314 ) );
[ 12, 46 ]
```

4.5 The ATLAS of Group Representations

This library is provided by the AtlasRep package.

4.5.1 IdClassNrToAtlasName

▷ IdClassNrToAtlasName(*k*, *i*) (function)

Returns: a string name such that AtlasGroup(name) is isomorphic to SmallClassNrGroup(*k*, *i*).

Example

```
gap> IdClassNrToAtlasName( 11, 34 );
"L2(17)"
gap> IdClassNr( AtlasGroup( "L2(17)" ) );
[ 11, 34 ]
```

Index