Prescribed Automorphism Groups: A GAP Package*

Vedran Krčadinac

University of Zagreb, Croatia

4th Croatian Combinatorial Days

Zagreb, September 22-23, 2022

* This work has been supported by the Croatian Science Foundation under the project 9752.

GAP

GAP is a computer algebra system for computational discrete algebra.

The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.12.0*, 2022. https://www.gap-system.org

GAP

GAP is a computer algebra system for computational discrete algebra.

The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.12.0*, 2022. https://www.gap-system.org

- Emphasis on Computational Group Theory

GAP

GAP is a computer algebra system for computational discrete algebra.

The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.12.0*, 2022. https://www.gap-system.org

- Emphasis on Computational Group Theory
- Programming language

GAP

GAP is a computer algebra system for computational discrete algebra.

The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.12.0*, 2022. https://www.gap-system.org

- Emphasis on Computational Group Theory
- Programming language
- Library of functions implementing algebraic algorithms

GAP

GAP is a computer algebra system for computational discrete algebra.

The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.12.0*, 2022. https://www.gap-system.org

- Emphasis on Computational Group Theory
- Programming language
- Library of functions implementing algebraic algorithms
- Data libraries of algebraic objects

GAP

GAP is a computer algebra system for computational discrete algebra.

The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.12.0*, 2022. https://www.gap-system.org

- Emphasis on Computational Group Theory
- Programming language
- Library of functions implementing algebraic algorithms
- Data libraries of algebraic objects
- Freely distributed for Linux, Windows and macOS
  https://www.gap-system.org/Releases/

GAP

GAP is a computer algebra system for computational discrete algebra.

The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.12.0*, 2022. https://www.gap-system.org

- Emphasis on Computational Group Theory
- Programming language
- Library of functions implementing algebraic algorithms
- Data libraries of algebraic objects
- Freely distributed for Linux, Windows and macOS
  https://www.gap-system.org/Releases/
- Packages to extend the mathematical capabilities of GAP

GAP packages

Some GAP packages of interest to combinatorialists:

GAP packages

Some GAP packages of interest to combinatorialists:

- **GRAPE** – A package for computing with graphs and groups.

GAP packages

Some GAP packages of interest to combinatorialists:

- **GRAPE** – A package for computing with graphs and groups.
- **Digraphs** – A package for digraphs and multidigraphs.

GAP packages

Some GAP packages of interest to combinatorialists:

- **GRAPE** – A package for computing with graphs and groups.
- **Digraphs** – A package for digraphs and multidigraphs.
- **Design** – A package for studying block designs.

GAP packages

Some GAP packages of interest to combinatorialists:

- **GRAPE** – A package for computing with graphs and groups.
- **Digraphs** – A package for digraphs and multidigraphs.
- **Design** – A package for studying block designs.
- **GUAVA** – A package for computing with codes.

GAP packages

Some GAP packages of interest to combinatorialists:

- **GRAPE** – A package for computing with graphs and groups.
- **Digraphs** – A package for digraphs and multidigraphs.
- **Design** – A package for studying block designs.
- **GUAVA** – A package for computing with codes.
- **AssociationSchemes** – A package for working with association schemes and homogeneous coherent configurations.

GAP packages

Some GAP packages of interest to combinatorialists:

- **GRAPE** – A package for computing with graphs and groups.
- **Digraphs** – A package for digraphs and multidigraphs.
- **Design** – A package for studying block designs.
- **GUAVA** – A package for computing with codes.
- **AssociationSchemes** – A package for working with association schemes and homogeneous coherent configurations.

Complete lists of GAP packages:

https://www.gap-system.org/Packages/packages.html

https://gap-packages.github.io/

The PAG package

PAG – Prescribed Automorphism Groups

- PAG is a GAP package for constructing combinatorial objects with prescribed automorphism groups.

The PAG package

PAG – Prescribed Automorphism Groups

- PAG is a GAP package for constructing combinatorial objects with prescribed automorphism groups.
- Development supported by the Croatian Science Foundation under the ACCO project (grant no. IP-2020-02-9752).

The PAG package

PAG – Prescribed Automorphism Groups

- PAG is a GAP package for constructing combinatorial objects with prescribed automorphism groups.
- Development supported by the Croatian Science Foundation under the ACCO project (grant no. IP-2020-02-9752).
- The PAG manual (version 0.1.2) is available at `https://web.math.pmf.unizg.hr/acco/publications.php`

The PAG package

PAG – Prescribed Automorphism Groups

- PAG is a GAP package for constructing combinatorial objects with prescribed automorphism groups.
- Development supported by the Croatian Science Foundation under the ACCO project (grant no. IP-2020-02-9752).
- The PAG manual (version 0.1.2) is available at `https://web.math.pmf.unizg.hr/acco/publications.php`
- The current installation files (version 0.1.4) can be obtained from the authors. Please write to vedran.krcadinac@math.hr.

PAG in action: Constructing block designs

A t-(v, k, λ) design is a v-element set V of points and a family $\mathcal{B}$ of k-subsets of V called blocks such that every t-subset of V is contained in exactly λ blocks.

PAG in action: Constructing block designs

A t-(v, k, λ) design is a v-element set V of points and a family $\mathcal{B}$ of k-subsets of V called blocks such that every t-subset of V is contained in exactly λ blocks.

An automorphism of the design is a permutation of V mapping blocks on blocks, i.e. preserving $\mathcal{B}$.

PAG in action: Constructing block designs

A t-(v, k, λ) design is a v-element set V of points and a family $\mathcal{B}$ of k-subsets of V called blocks such that every t-subset of V is contained in exactly λ blocks.

An automorphism of the design is a permutation of V mapping blocks on blocks, i.e. preserving $\mathcal{B}$.

Goal:

Construct t-(v, k, λ) designs with a prescribed group of automorphisms G.

PAG in action: Constructing block designs

A t-(v, k, λ) design is a v-element set V of points and a family $\mathcal{B}$ of k-subsets of V called blocks such that every t-subset of V is contained in exactly λ blocks.

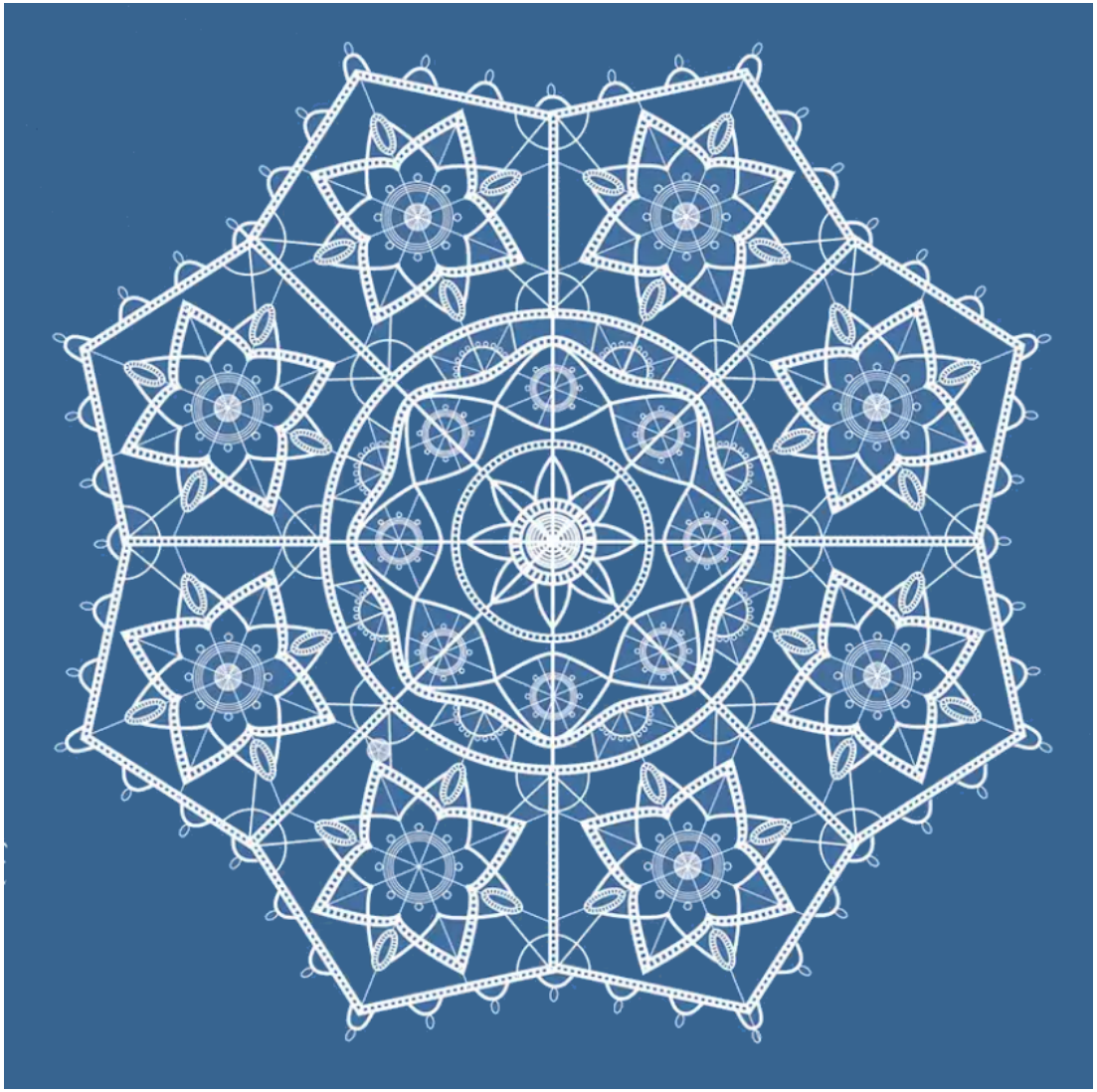
An automorphism of the design is a permutation of V mapping blocks on blocks, i.e. preserving $\mathcal{B}$.

Goal:

Construct t-(v, k, λ) designs with a prescribed group of automorphisms G.

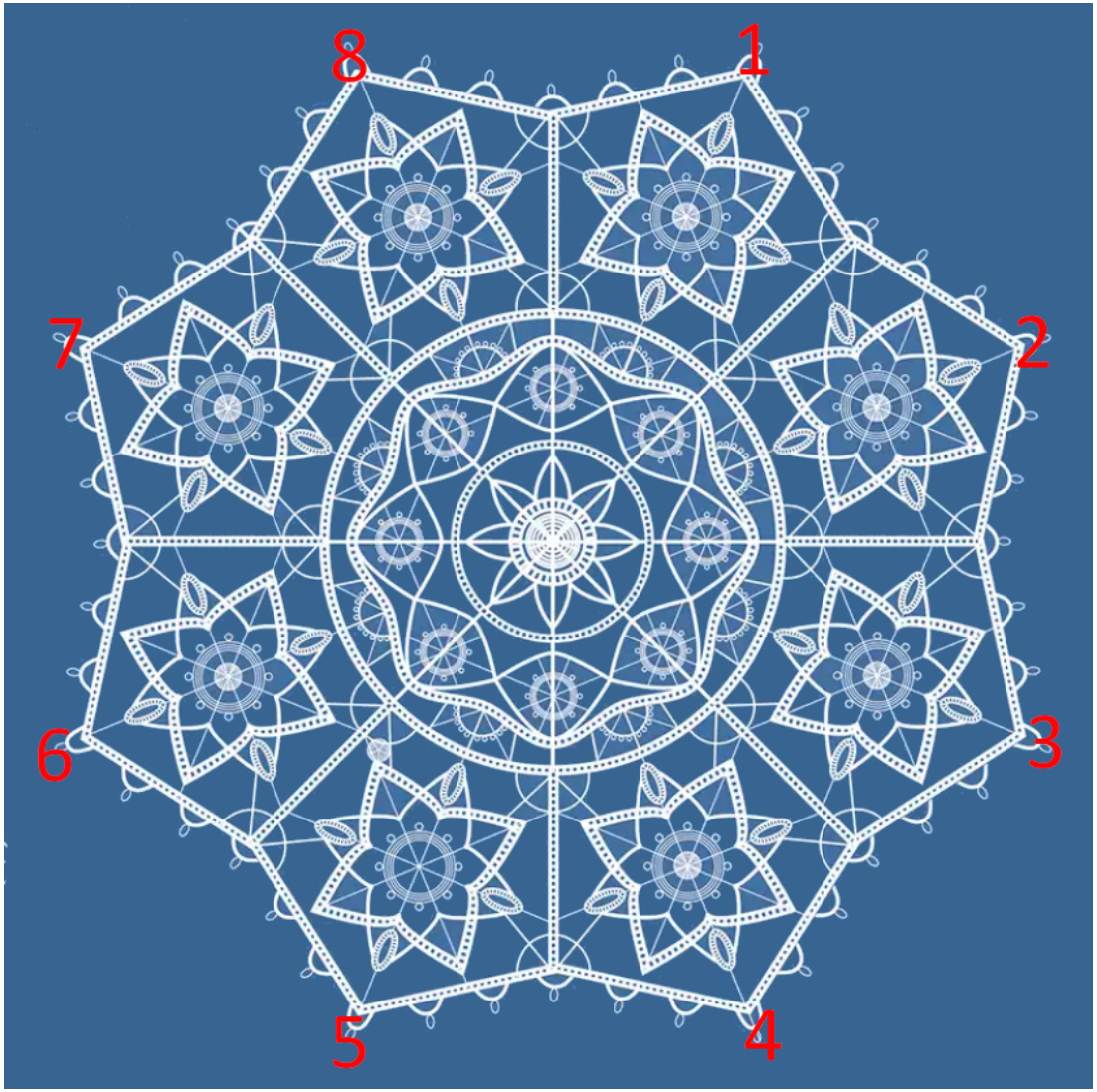
How to choose G ?

PAG in action: Constructing block designs



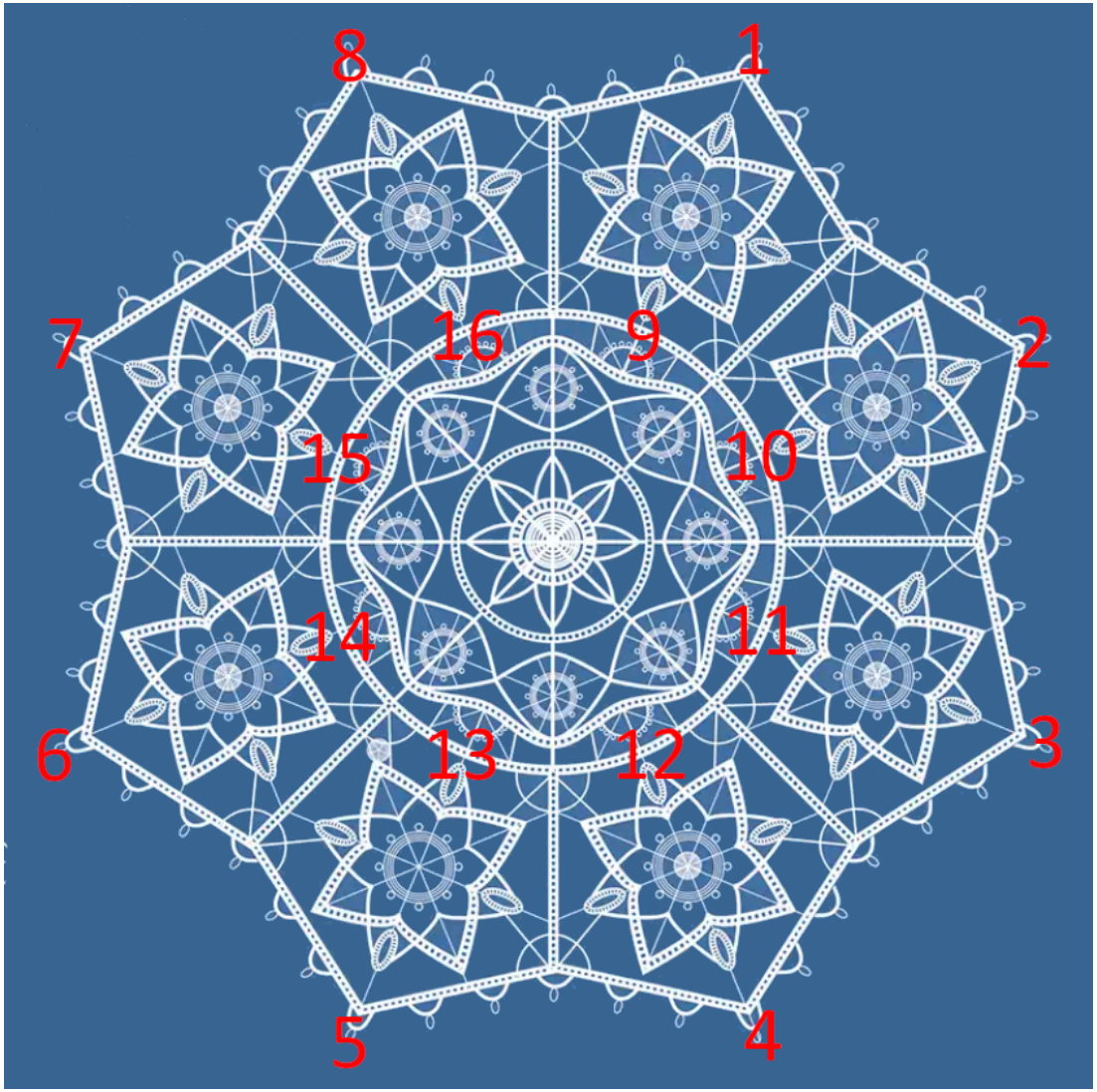
Source: https://www.plakati.com.hr

PAG in action: Constructing block designs



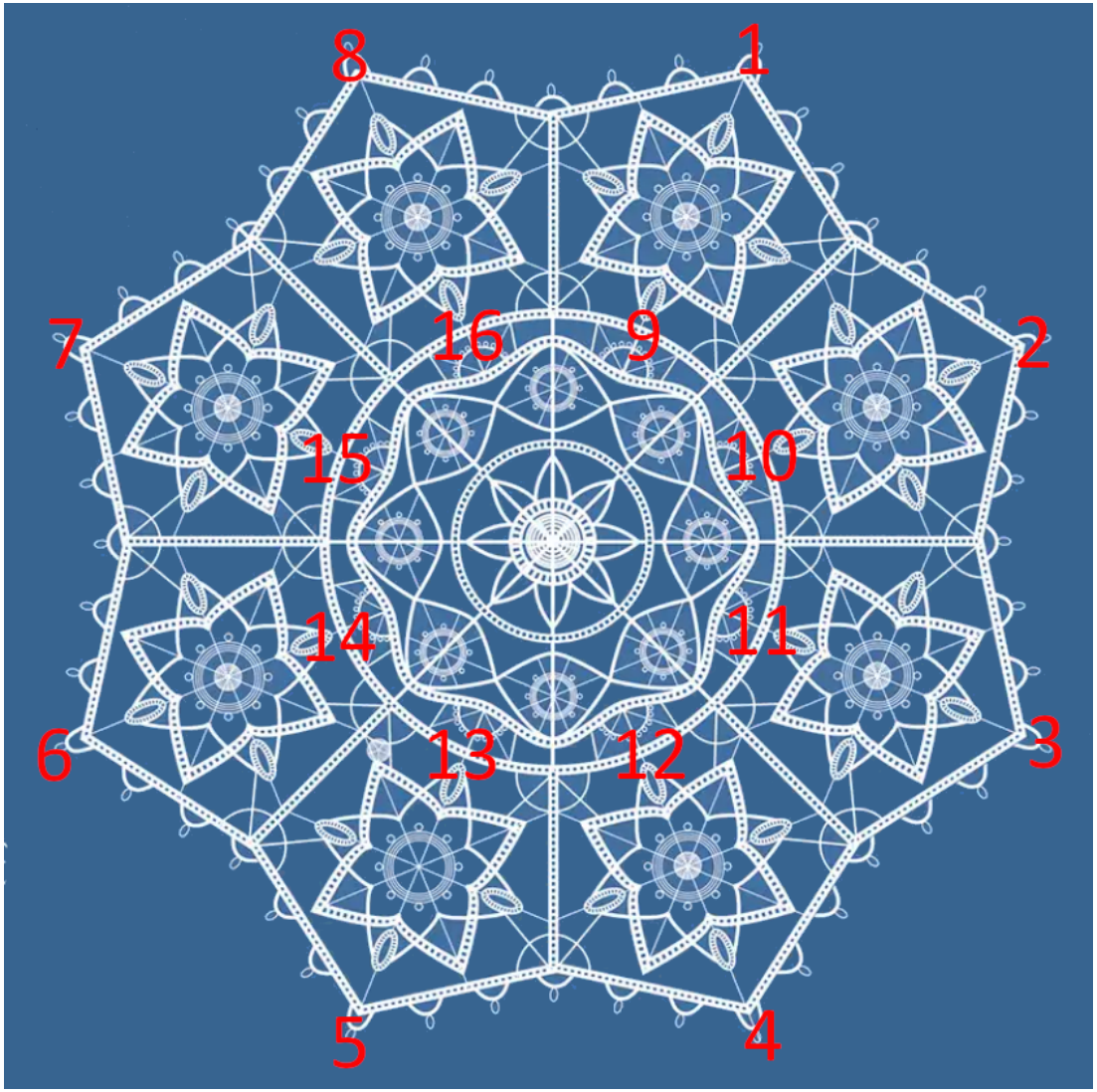
Source: https://www.plakati.com.hr

PAG in action: Constructing block designs



Source: https://www.plakati.com.hr

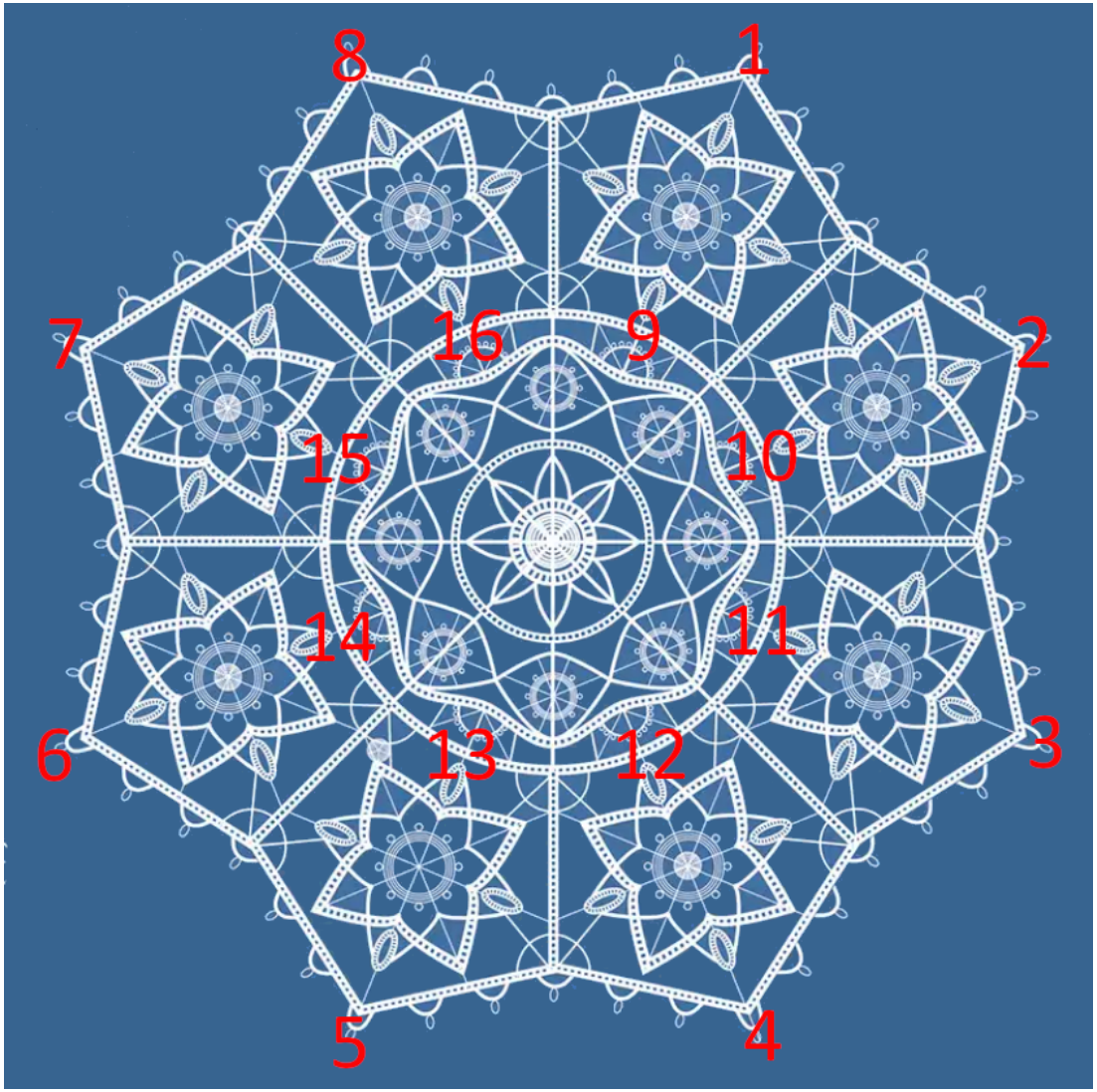
PAG in action: Constructing block designs




```
a:=(1,2,3,4,5,6,7,8)
(9,10,11,12,13,14,15,16);
```

Source: https://www.plakati.com.hr

PAG in action: Constructing block designs

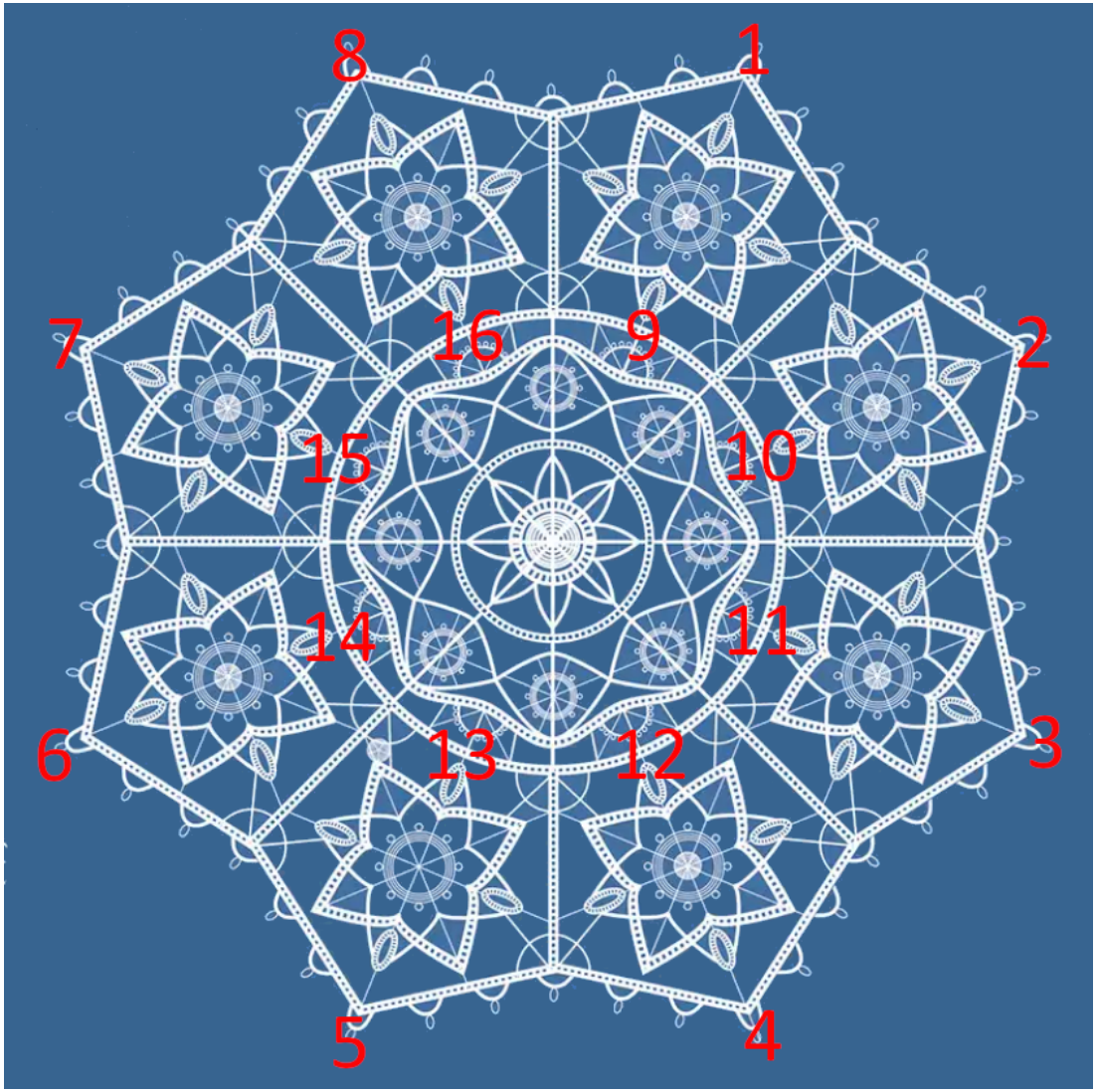


Source: https://www.plakati.com.hr

```
a:=(1,2,3,4,5,6,7,8)
(9,10,11,12,13,14,15,16);

b:=(1,8)(2,7)(3,6)(4,5)
(9,16)(10,15)(11,14)(12,13);
```

PAG in action: Constructing block designs



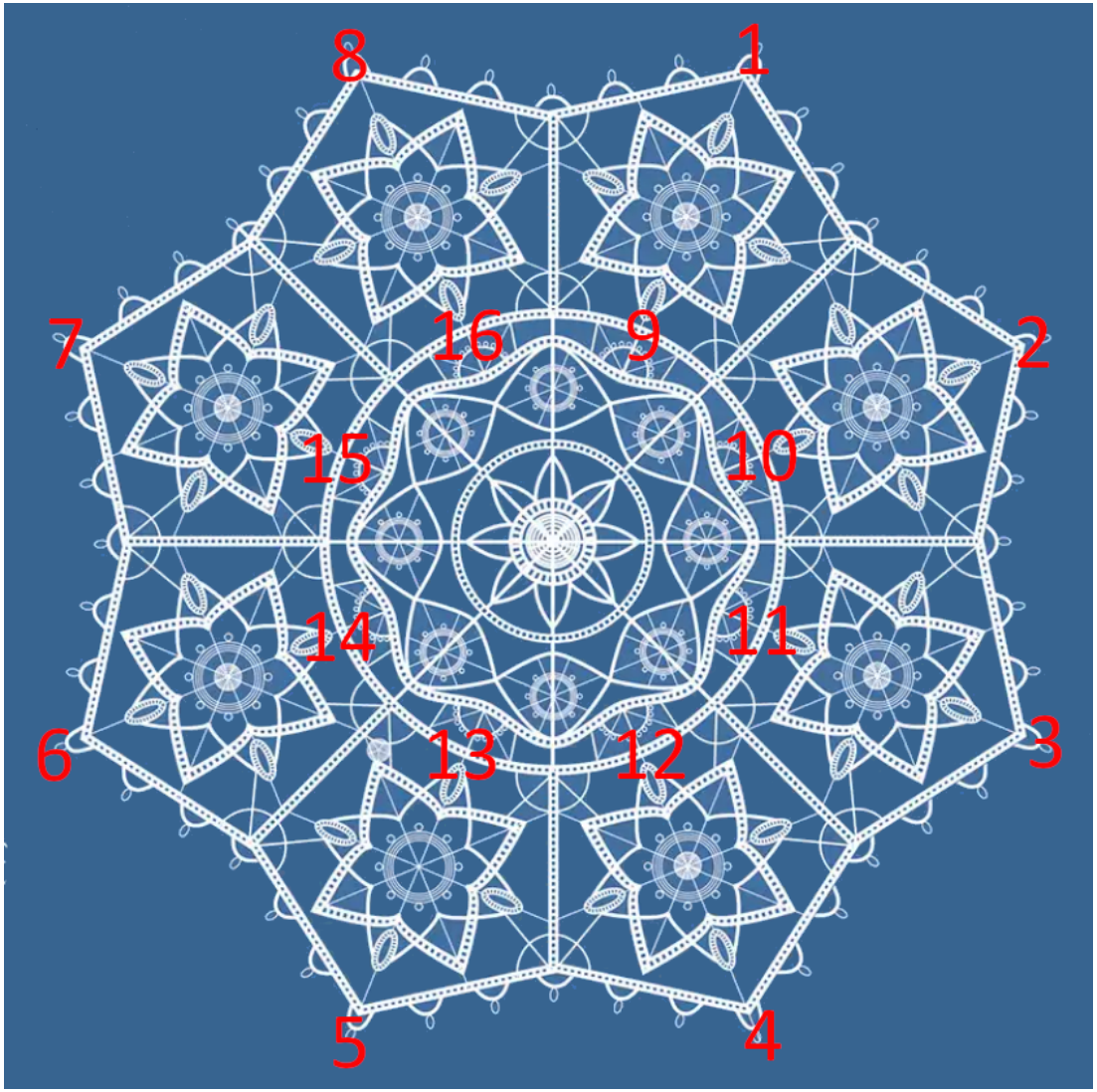
Source: https://www.plakati.com.hr

```
a:=(1,2,3,4,5,6,7,8)
(9,10,11,12,13,14,15,16);

b:=(1,8)(2,7)(3,6)(4,5)
(9,16)(10,15)(11,14)(12,13);

g:=Group(a,b);
```

PAG in action: Constructing block designs



Source: https://www.plakati.com.hr

```
a:=(1,2,3,4,5,6,7,8)
(9,10,11,12,13,14,15,16);

b:=(1,8)(2,7)(3,6)(4,5)
(9,16)(10,15)(11,14)(12,13);

g:=Group(a,b);

StructureDescription(g);
"D16"
```

PAG in action: Constructing block designs

A t-(v, k, λ) design is a v-element set V of points and a family $\mathcal{B}$ of k-subsets of V called blocks such that every t-subset of V is contained in exactly λ blocks.

An automorphism of the design is a permutation of V mapping blocks onto blocks, i.e. preserving $\mathcal{B}$.

Goal:

Construct t-(v, k, λ) designs with a prescribed group of automorphisms G.

How to choose G ? $\rightsquigarrow$ D_{16} acting on 16 points

PAG in action: Constructing block designs

A t-(v, k, λ) design is a v-element set V of points and a family $\mathcal{B}$ of k-subsets of V called blocks such that every t-subset of V is contained in exactly λ blocks.

An automorphism of the design is a permutation of V mapping blocks onto blocks, i.e. preserving $\mathcal{B}$.

Goal:

Construct t-(v, k, λ) designs with a prescribed group of automorphisms G.

How to choose G ? $\rightsquigarrow$ D_{16} acting on 16 points

Suitable design parameters?

PAG in action: Constructing block designs

A t-(v, k, λ) design is a v-element set V of points and a family $\mathcal{B}$ of k-subsets of V called blocks such that every t-subset of V is contained in exactly λ blocks.

An automorphism of the design is a permutation of V mapping blocks onto blocks, i.e. preserving $\mathcal{B}$.

Goal:

Construct t-(v, k, λ) designs with a prescribed group of automorphisms G.

How to choose G ? $\rightsquigarrow$ D_{16} acting on 16 points

Suitable design parameters?

- 2-$(16, 6, 2)$ $\rightsquigarrow$ second smallest biplanes
- 2-$(16, 4, 1)$ $\rightsquigarrow$ affine plane of order 4
- 3-$(16, 4, 1)$ $\rightsquigarrow$ Steiner quadruple systems
- 5-$(16, 6, 3)$ $\rightsquigarrow$ 5-designs

A GAP session

```
âââââââââ  GAP 4.11.1 of 2021-03-02
â  GAP  â   https://www.gap-system.org
âââââââââ   Architecture: x86_64-pc-linux-gnu-default64-kv7
Configuration:  gmp 6.2.0, GASMAN, readline
Loading the library and packages ...
Packages:   AClib 1.3.2, Alnuth 3.1.2, AtlasRep 2.1.0, AutoDoc 2020.08.11, AutPGrp
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap>
```

A GAP session

```
âââââââââ  GAP 4.11.1 of 2021-03-02
â  GAP  â   https://www.gap-system.org
âââââââââ   Architecture: x86_64-pc-linux-gnu-default64-kv7
Configuration:  gmp 6.2.0, GASMAN, readline
Loading the library and packages ...
Packages:   AClib 1.3.2, Alnuth 3.1.2, AtlasRep 2.1.0, AutoDoc 2020.08.11, AutPGrp
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap> LoadPackage("PAG");
```

A GAP session

```
ââââââââââ  GAP 4.11.1 of 2021-03-02
â   GAP   â   https://www.gap-system.org
ââââââââââ   Architecture: x86_64-pc-linux-gnu-default64-kv7
Configuration:  gmp 6.2.0, GASMAN, readline
Loading the library and packages ...
Packages:   AClib 1.3.2, Alnuth 3.1.2, AtlasRep 2.1.0, AutoDoc 2020.08.11, AutPGrp
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap> LoadPackage("PAG");
ââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
Loading  PAG 0.1.4 (Prescribed Automorphism Groups)
by Vedran Krcadinac (https://web.math.pmf.unizg.hr/~krcko/homepage.html).
Homepage: https://gap-packages.github.io/pag
Report issues at https://github.com/gap-packages/pag/issues
ââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
true
gap>
```

A GAP session

```
ââââââââââ  GAP 4.11.1 of 2021-03-02
â   GAP   â   https://www.gap-system.org
ââââââââââ   Architecture: x86_64-pc-linux-gnu-default64-kv7
Configuration:  gmp 6.2.0, GASMAN, readline
Loading the library and packages ...
Packages:   AClib 1.3.2, Alnuth 3.1.2, AtlasRep 2.1.0, AutoDoc 2020.08.11, AutPGrp
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap> LoadPackage("PAG");
âââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
Loading  PAG 0.1.4 (Prescribed Automorphism Groups)
by Vedran Krcadinac (https://web.math.pmf.unizg.hr/~krcko/homepage.html).
Homepage: https://gap-packages.github.io/pag
Report issues at https://github.com/gap-packages/pag/issues
âââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
true
gap> a:=(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16);
```

A GAP session

```
âââââââââ  GAP 4.11.1 of 2021-03-02
â  GAP  â   https://www.gap-system.org
âââââââââ   Architecture: x86_64-pc-linux-gnu-default64-kv7
Configuration:  gmp 6.2.0, GASMAN, readline
Loading the library and packages ...
Packages:   AClib 1.3.2, Alnuth 3.1.2, AtlasRep 2.1.0, AutoDoc 2020.08.11, AutPGrp
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap> LoadPackage("PAG");
ââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
Loading  PAG 0.1.4 (Prescribed Automorphism Groups)
by Vedran Krcadinac (https://web.math.pmf.unizg.hr/~krcko/homepage.html).
Homepage: https://gap-packages.github.io/pag
Report issues at https://github.com/gap-packages/pag/issues
ââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
true
gap> a:=(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16);
(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16)
gap>
```

A GAP session

```
ââââââââââ  GAP 4.11.1 of 2021-03-02
â   GAP   â   https://www.gap-system.org
ââââââââââ   Architecture: x86_64-pc-linux-gnu-default64-kv7
Configuration:  gmp 6.2.0, GASMAN, readline
Loading the library and packages ...
Packages:   AClib 1.3.2, Alnuth 3.1.2, AtlasRep 2.1.0, AutoDoc 2020.08.11, AutPGrp
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap> LoadPackage("PAG");
âââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
Loading  PAG 0.1.4 (Prescribed Automorphism Groups)
by Vedran Krcadinac (https://web.math.pmf.unizg.hr/~krcko/homepage.html).
Homepage: https://gap-packages.github.io/pag
Report issues at https://github.com/gap-packages/pag/issues
âââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
true
gap> a:=(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16);
(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16)
gap> b:=(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13);
```

A GAP session

```
ââââââââââ  GAP 4.11.1 of 2021-03-02
â   GAP   â   https://www.gap-system.org
ââââââââââ   Architecture: x86_64-pc-linux-gnu-default64-kv7
Configuration:  gmp 6.2.0, GASMAN, readline
Loading the library and packages ...
Packages:   AClib 1.3.2, Alnuth 3.1.2, AtlasRep 2.1.0, AutoDoc 2020.08.11, AutPGrp
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap> LoadPackage("PAG");
ââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
Loading  PAG 0.1.4 (Prescribed Automorphism Groups)
by Vedran Krcadinac (https://web.math.pmf.unizg.hr/~krcko/homepage.html).
Homepage: https://gap-packages.github.io/pag
Report issues at https://github.com/gap-packages/pag/issues
ââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
true
gap> a:=(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16);
(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16)
gap> b:=(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13);
(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13)
gap>
```

A GAP session

```
âââââââââ   GAP 4.11.1 of 2021-03-02
â   GAP   â   https://www.gap-system.org
âââââââââ   Architecture: x86_64-pc-linux-gnu-default64-kv7
Configuration:  gmp 6.2.0, GASMAN, readline
Loading the library and packages ...
Packages:   AClib 1.3.2, Alnuth 3.1.2, AtlasRep 2.1.0, AutoDoc 2020.08.11, AutPGrp
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap> LoadPackage("PAG");
ââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
Loading  PAG 0.1.4 (Prescribed Automorphism Groups)
by Vedran Krcadinac (https://web.math.pmf.unizg.hr/~krcko/homepage.html).
Homepage: https://gap-packages.github.io/pag
Report issues at https://github.com/gap-packages/pag/issues
ââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
true
gap> a:=(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16);
(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16)
gap> b:=(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13);
(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13)
gap> g:=Group(a,b);
```

A GAP session

```
â  GAP  â   https://www.gap-system.org
âââââââââ   Architecture: x86_64-pc-linux-gnu-default64-kv7
Configuration:  gmp 6.2.0, GASMAN, readline
Loading the library and packages ...
Packages:   AClib 1.3.2, Alnuth 3.1.2, AtlasRep 2.1.0, AutoDoc 2020.08.11, AutPGrp
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap> LoadPackage("PAG");
ââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
Loading  PAG 0.1.4 (Prescribed Automorphism Groups)
by Vedran Krcadinac (https://web.math.pmf.unizg.hr/~krcko/homepage.html).
Homepage: https://gap-packages.github.io/pag
Report issues at https://github.com/gap-packages/pag/issues
ââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
true
gap> a:=(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16);
(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16)
gap> b:=(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13);
(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13)
gap> g:=Group(a,b);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16), (1,8)(2,7)(3,6)(4,5)(9,16)(10,15
gap>
```

A GAP session

```
â  GAP  â   https://www.gap-system.org
âââââââââ   Architecture: x86_64-pc-linux-gnu-default64-kv7
Configuration:  gmp 6.2.0, GASMAN, readline
Loading the library and packages ...
Packages:   AClib 1.3.2, Alnuth 3.1.2, AtlasRep 2.1.0, AutoDoc 2020.08.11, AutPGrp
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap> LoadPackage("PAG");
ââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
Loading  PAG 0.1.4 (Prescribed Automorphism Groups)
by Vedran Krcadinac (https://web.math.pmf.unizg.hr/~krcko/homepage.html).
Homepage: https://gap-packages.github.io/pag
Report issues at https://github.com/gap-packages/pag/issues
ââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
true
gap> a:=(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16);
(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16)
gap> b:=(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13);
(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13)
gap> g:=Group(a,b);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16), (1,8)(2,7)(3,6)(4,5)(9,16)(10,15
gap> StructureDescription(g);
```

A GAP session

```
Configuration:  gmp 6.2.0, GASMAN, readline
Loading the library and packages ...
Packages:   AClib 1.3.2, Alnuth 3.1.2, AtlasRep 2.1.0, AutoDoc 2020.08.11, AutPGrp
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap> LoadPackage("PAG");
âââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
Loading  PAG 0.1.4 (Prescribed Automorphism Groups)
by Vedran Krcadinac (https://web.math.pmf.unizg.hr/~krcko/homepage.html).
Homepage: https://gap-packages.github.io/pag
Report issues at https://github.com/gap-packages/pag/issues
âââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
true
gap> a:=(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16);
(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16)
gap> b:=(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13);
(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13)
gap> g:=Group(a,b);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16), (1,8)(2,7)(3,6)(4,5)(9,16)(10,15
gap> StructureDescription(g);
"D16"
gap>
```

A GAP session

```
Configuration:  gmp 6.2.0, GASMAN, readline
Loading the library and packages ...
Packages:   AClib 1.3.2, Alnuth 3.1.2, AtlasRep 2.1.0, AutoDoc 2020.08.11, AutPGrp
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap> LoadPackage("PAG");
âââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
Loading  PAG 0.1.4 (Prescribed Automorphism Groups)
by Vedran Krcadinac (https://web.math.pmf.unizg.hr/~krcko/homepage.html).
Homepage: https://gap-packages.github.io/pag
Report issues at https://github.com/gap-packages/pag/issues
âââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââââ
true
gap> a:=(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16);
(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16)
gap> b:=(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13);
(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13)
gap> g:=Group(a,b);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16), (1,8)(2,7)(3,6)(4,5)(9,16)(10,15
gap> StructureDescription(g);
"D16"
gap> KramerMesnerSearch(2,16,6,2,g);
```

A GAP session

```
            CTblLib 1.3.1, FactInt 1.6.3, FGA 1.4.0, Forms 1.2.5, GAPDoc 1.6.4, gen
            PrimGrp 3.4.1, RadiRoot 2.8, recog 1.3.2, ResClasses 4.7.2, SmallGrp 1.
Try '??help' for help. See also '?copyright', '?cite' and '?authors'
gap> LoadPackage("PAG");
ªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªª
Loading  PAG 0.1.4 (Prescribed Automorphism Groups)
by Vedran Krcadinac (https://web.math.pmf.unizg.hr/~krcko/homepage.html).
Homepage: https://gap-packages.github.io/pag
Report issues at https://github.com/gap-packages/pag/issues
ªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªªª
true
gap> a:=(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16);
(1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16)
gap> b:=(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13);
(1,8)(2,7)(3,6)(4,5)(9,16)(10,15)(11,14)(12,13)
gap> g:=Group(a,b);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16), (1,8)(2,7)(3,6)(4,5)(9,16)(10,15
gap> StructureDescription(g);
"D16"
gap> KramerMesnerSearch(2,16,6,2,g);
Computing t-subset orbit representatives...
13
Computing k-subset orbit representatives...
```

A GAP session

```
0001000000000000000001000000000
0000000000010000000001000000000
0001000000000000000000100000000
0000000000010000000000100000000
1000000000000000000000010000000
0000000001000000000000010000000
1000000000000000000000001000000
0000000001000000000000001000000
Prune_cs: 43
Prune_only_zeros: 0 of 42
Prune_hoelder: 0 of 42
Prune_N: 0
Fincke-Pohst: 0
Loops: 93
Total number of solutions: 8

total enumeration time: 0:00:00
[ [ [ 1, 2, 3, 6, 13, 15 ], [ 1, 3, 9, 10, 11, 14 ] ], [ [ 1, 2, 3, 6, 13, 15 ], [
  [ [ 1, 2, 3, 6, 13, 15 ], [ 1, 3, 9, 10, 11, 14 ] ], [ [ 1, 2, 4, 5, 14, 16 ], [
  [ [ 1, 2, 4, 5, 14, 16 ], [ 1, 3, 9, 11, 12, 16 ] ], [ [ 1, 2, 3, 6, 9, 11 ], [ 1
  [ [ 1, 2, 3, 6, 9, 11 ], [ 1, 3, 12, 13, 15, 16 ] ], [ [ 1, 2, 4, 5, 10, 12 ], [
gap>
```

A GAP session

```
0001000000000000000001000000000
0000000000010000000001000000000
0001000000000000000000100000000
0000000000010000000000100000000
1000000000000000000000010000000
0000000001000000000000010000000
1000000000000000000000001000000
0000000001000000000000001000000
Prune_cs: 43
Prune_only_zeros: 0 of 42
Prune_hoelder: 0 of 42
Prune_N: 0
Fincke-Pohst: 0
Loops: 93
Total number of solutions: 8

total enumeration time: 0:00:00
[ [ [ 1, 2, 3, 6, 13, 15 ], [ 1, 3, 9, 10, 11, 14 ] ], [ [ 1, 2, 3, 6, 13, 15 ], [
  [ [ 1, 2, 3, 6, 13, 15 ], [ 1, 3, 9, 10, 11, 14 ] ], [ [ 1, 2, 4, 5, 14, 16 ], [
  [ [ 1, 2, 4, 5, 14, 16 ], [ 1, 3, 9, 11, 12, 16 ] ], [ [ 1, 2, 3, 6, 9, 11 ], [ 1
  [ [ 1, 2, 3, 6, 9, 11 ], [ 1, 3, 12, 13, 15, 16 ] ], [ [ 1, 2, 4, 5, 10, 12 ], [
gap> d:=KramerMesnerSearch(2,16,6,2,g,rec(NonIsomorphic:=true));
```

A GAP session

```
0000000001000000000000010000000
1000000000000000000000001000000
0000000001000000000000001000000
Prune_cs: 43
Prune_only_zeros: 0 of 42
Prune_hoelder: 0 of 42
Prune_N: 0
Fincke-Pohst: 0
Loops: 93
Total number of solutions: 8

total enumeration time: 0:00:00
Performing isomorph rejection...
2
[ rec( autGroup := Group([ (1,5)(3,7)(9,13)(11,15), (1,2)(3,12,9,8,15,10)(4,11,14,7
      autSubgroup := D16, blocks := [ [ 1, 2, 4, 5, 14, 16 ], [ 1, 2, 6, 7, 11, 13 ]
          [ 1, 7, 9, 12, 15, 16 ], [ 2, 3, 5, 6, 9, 15 ], [ 2, 3, 7, 8, 12, 14 ], [
          [ 3, 5, 11, 12, 13, 16 ], [ 4, 5, 7, 8, 9, 11 ], [ 4, 6, 9, 12, 13, 14 ],
      v := 16 ), rec( autGroup := <permutation group with 6 generators>, autSubgrou
          [ 1, 4, 5, 6, 10, 16 ], [ 1, 4, 7, 8, 11, 13 ], [ 1, 7, 9, 12, 15, 16 ],
          [ 2, 5, 6, 7, 9, 11 ], [ 2, 8, 9, 10, 13, 16 ], [ 3, 4, 5, 8, 9, 15 ], [
          [ 6, 8, 11, 14, 15, 16 ] ], isBinary := true, isBlockDesign := true, v :=
gap>
```

A GAP session

```
00000000010000000000000010000000
10000000000000000000000001000000
00000000010000000000000001000000
Prune_cs: 43
Prune_only_zeros: 0 of 42
Prune_hoelder: 0 of 42
Prune_N: 0
Fincke-Pohst: 0
Loops: 93
Total number of solutions: 8

total enumeration time: 0:00:00
Performing isomorph rejection...
2
[ rec( autGroup := Group([ (1,5)(3,7)(9,13)(11,15), (1,2)(3,12,9,8,15,10)(4,11,14,7
      autSubgroup := D16, blocks := [ [ 1, 2, 4, 5, 14, 16 ], [ 1, 2, 6, 7, 11, 13 ]
          [ 1, 7, 9, 12, 15, 16 ], [ 2, 3, 5, 6, 9, 15 ], [ 2, 3, 7, 8, 12, 14 ], [
          [ 3, 5, 11, 12, 13, 16 ], [ 4, 5, 7, 8, 9, 11 ], [ 4, 6, 9, 12, 13, 14 ],
      v := 16 ), rec( autGroup := <permutation group with 6 generators>, autSubgrou
          [ 1, 4, 5, 6, 10, 16 ], [ 1, 4, 7, 8, 11, 13 ], [ 1, 7, 9, 12, 15, 16 ],
          [ 2, 5, 6, 7, 9, 11 ], [ 2, 8, 9, 10, 13, 16 ], [ 3, 4, 5, 8, 9, 15 ], [
          [ 6, 8, 11, 14, 15, 16 ] ], isBinary := true, isBlockDesign := true, v :=
gap> List(d,AllTDesignLambdas);
```

A GAP session

```
00000000010000000000000001000000
Prune_cs: 43
Prune_only_zeros: 0 of 42
Prune_hoelder: 0 of 42
Prune_N: 0
Fincke-Pohst: 0
Loops: 93
Total number of solutions: 8

total enumeration time: 0:00:00
Performing isomorph rejection...
2
[ rec( autGroup := Group([ (1,5)(3,7)(9,13)(11,15), (1,2)(3,12,9,8,15,10)(4,11,14,7
      autSubgroup := D16, blocks := [ [ 1, 2, 4, 5, 14, 16 ], [ 1, 2, 6, 7, 11, 13 ]
          [ 1, 7, 9, 12, 15, 16 ], [ 2, 3, 5, 6, 9, 15 ], [ 2, 3, 7, 8, 12, 14 ], [
          [ 3, 5, 11, 12, 13, 16 ], [ 4, 5, 7, 8, 9, 11 ], [ 4, 6, 9, 12, 13, 14 ],
      v := 16 ), rec( autGroup := <permutation group with 6 generators>, autSubgrou
          [ 1, 4, 5, 6, 10, 16 ], [ 1, 4, 7, 8, 11, 13 ], [ 1, 7, 9, 12, 15, 16 ],
          [ 2, 5, 6, 7, 9, 11 ], [ 2, 8, 9, 10, 13, 16 ], [ 3, 4, 5, 8, 9, 15 ], [
          [ 6, 8, 11, 14, 15, 16 ] ], isBinary := true, isBlockDesign := true, v :=
gap> List(d,AllTDesignLambdas);
[ [ 16, 6, 2 ], [ 16, 6, 2 ] ]
gap>
```

A GAP session

```
00000000010000000000000001000000
Prune_cs: 43
Prune_only_zeros: 0 of 42
Prune_hoelder: 0 of 42
Prune_N: 0
Fincke-Pohst: 0
Loops: 93
Total number of solutions: 8

total enumeration time: 0:00:00
Performing isomorph rejection...
2
[ rec( autGroup := Group([ (1,5)(3,7)(9,13)(11,15), (1,2)(3,12,9,8,15,10)(4,11,14,7
      autSubgroup := D16, blocks := [ [ 1, 2, 4, 5, 14, 16 ], [ 1, 2, 6, 7, 11, 13 ]
          [ 1, 7, 9, 12, 15, 16 ], [ 2, 3, 5, 6, 9, 15 ], [ 2, 3, 7, 8, 12, 14 ], [
          [ 3, 5, 11, 12, 13, 16 ], [ 4, 5, 7, 8, 9, 11 ], [ 4, 6, 9, 12, 13, 14 ],
      v := 16 ), rec( autGroup := <permutation group with 6 generators>, autSubgrou
          [ 1, 4, 5, 6, 10, 16 ], [ 1, 4, 7, 8, 11, 13 ], [ 1, 7, 9, 12, 15, 16 ],
          [ 2, 5, 6, 7, 9, 11 ], [ 2, 8, 9, 10, 13, 16 ], [ 3, 4, 5, 8, 9, 15 ], [
          [ 6, 8, 11, 14, 15, 16 ] ], isBinary := true, isBlockDesign := true, v :=
gap> List(d,AllTDesignLambdas);
[ [ 16, 6, 2 ], [ 16, 6, 2 ] ]
gap> d:=KramerMesnerSearch(2,16,4,1,g,rec(NonIsomorphic:=true));
```

A GAP session

```
Dimension of solution space (k): 2 compared to s-z+2: -5
Number of nonzero entries in the last row: 1
Fq: 1.000000
Fd: 9.009000
16.000 13.000


0 1
0 1


Prune_cs: 1
Prune_only_zeros: 0 of 0
Prune_hoelder: 0 of 0
Prune_N: 0
Fincke-Pohst: 0
Loops: 1
Total number of solutions: 0

total enumeration time: 0:00:00
Performing isomorph rejection...
0
gap>
```

A GAP session

```
Dimension of solution space (k): 2 compared to s-z+2: -5
Number of nonzero entries in the last row: 1
Fq: 1.000000
Fd: 9.009000
16.000 13.000


0 1
0 1


Prune_cs: 1
Prune_only_zeros: 0 of 0
Prune_hoelder: 0 of 0
Prune_N: 0
Fincke-Pohst: 0
Loops: 1
Total number of solutions: 0

total enumeration time: 0:00:00
Performing isomorph rejection...
0
gap> g2:=Group(a);
```

A GAP session

```
Fq: 1.000000
Fd: 9.009000
16.000 13.000


0 1
0 1


Prune_cs: 1
Prune_only_zeros: 0 of 0
Prune_hoelder: 0 of 0
Prune_N: 0
Fincke-Pohst: 0
Loops: 1
Total number of solutions: 0

total enumeration time: 0:00:00
Performing isomorph rejection...
0
gap> g2:=Group(a);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16) ])
gap>
```

A GAP session

```
Fq: 1.000000
Fd: 9.009000
16.000 13.000



0 1
0 1



Prune_cs: 1
Prune_only_zeros: 0 of 0
Prune_hoelder: 0 of 0
Prune_N: 0
Fincke-Pohst: 0
Loops: 1
Total number of solutions: 0

total enumeration time: 0:00:00
Performing isomorph rejection...
0
gap> g2:=Group(a);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16) ])
gap> StructureDescription(g2);
```

A GAP session

```
16.000 13.000



0 1
0 1



Prune_cs: 1
Prune_only_zeros: 0 of 0
Prune_hoelder: 0 of 0
Prune_N: 0
Fincke-Pohst: 0
Loops: 1
Total number of solutions: 0

total enumeration time: 0:00:00
Performing isomorph rejection...
0
gap> g2:=Group(a);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16) ])
gap> StructureDescription(g2);
"C8"
gap>
```

A GAP session

```
16.000 13.000



0 1
0 1



Prune_cs: 1
Prune_only_zeros: 0 of 0
Prune_hoelder: 0 of 0
Prune_N: 0
Fincke-Pohst: 0
Loops: 1
Total number of solutions: 0

total enumeration time: 0:00:00
Performing isomorph rejection...
0
gap> g2:=Group(a);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16) ])
gap> StructureDescription(g2);
"C8"
gap> d:=KramerMesnerSearch(2,16,4,1,g2,rec(NonIsomorphic:=true));
```

A GAP session

```
000010000000000000000000000000000000000000000000000000000000000010000001000000000
000000000100000000000000000000000000000000000000000000000000000001000000000000000
Prune_cs: 792
Prune_only_zeros: 70 of 948
Prune_hoelder: 86 of 878
Prune_N: 0
Fincke-Pohst: 0
Loops: 1756
Total number of solutions: 16

total enumeration time: 0:00:00
Performing isomorph rejection...
1
[ rec( autGroup := <permutation group with 6 generators>, autSubgroup := C8,
      blocks := [ [ 1, 2, 7, 14 ], [ 1, 3, 4, 16 ], [ 1, 5, 11, 15 ],
          [ 1, 6, 8, 13 ], [ 1, 9, 10, 12 ], [ 2, 3, 8, 15 ], [ 2, 4, 5, 9 ],
          [ 2, 6, 12, 16 ], [ 2, 10, 11, 13 ], [ 3, 5, 6, 10 ],
          [ 3, 7, 9, 13 ], [ 3, 11, 12, 14 ], [ 4, 6, 7, 11 ],
          [ 4, 8, 10, 14 ], [ 4, 12, 13, 15 ], [ 5, 7, 8, 12 ],
          [ 5, 13, 14, 16 ], [ 6, 9, 14, 15 ], [ 7, 10, 15, 16 ],
          [ 8, 9, 11, 16 ] ], isBinary := true, isBlockDesign := true,
      v := 16 ) ]
gap>
```

A GAP session

```
000010000000000000000000000000000000000000000000000000000000000010000001000000000
000000000100000000000000000000000000000000000000000000000000000001000000000000000
Prune_cs: 792
Prune_only_zeros: 70 of 948
Prune_hoelder: 86 of 878
Prune_N: 0
Fincke-Pohst: 0
Loops: 1756
Total number of solutions: 16

total enumeration time: 0:00:00
Performing isomorph rejection...
1
[ rec( autGroup := <permutation group with 6 generators>, autSubgroup := C8,
      blocks := [ [ 1, 2, 7, 14 ], [ 1, 3, 4, 16 ], [ 1, 5, 11, 15 ],
          [ 1, 6, 8, 13 ], [ 1, 9, 10, 12 ], [ 2, 3, 8, 15 ], [ 2, 4, 5, 9 ],
          [ 2, 6, 12, 16 ], [ 2, 10, 11, 13 ], [ 3, 5, 6, 10 ],
          [ 3, 7, 9, 13 ], [ 3, 11, 12, 14 ], [ 4, 6, 7, 11 ],
          [ 4, 8, 10, 14 ], [ 4, 12, 13, 15 ], [ 5, 7, 8, 12 ],
          [ 5, 13, 14, 16 ], [ 6, 9, 14, 15 ], [ 7, 10, 15, 16 ],
          [ 8, 9, 11, 16 ] ], isBinary := true, isBlockDesign := true,
      v := 16 ) ]
gap> List(d,AllTDesignLambdas);
```

A GAP session

```
Prune_cs: 792
Prune_only_zeros: 70 of 948
Prune_hoelder: 86 of 878
Prune_N: 0
Fincke-Pohst: 0
Loops: 1756
Total number of solutions: 16

total enumeration time: 0:00:00
Performing isomorph rejection...
1
[ rec( autGroup := <permutation group with 6 generators>, autSubgroup := C8,
      blocks := [ [ 1, 2, 7, 14 ], [ 1, 3, 4, 16 ], [ 1, 5, 11, 15 ],
          [ 1, 6, 8, 13 ], [ 1, 9, 10, 12 ], [ 2, 3, 8, 15 ], [ 2, 4, 5, 9 ],
          [ 2, 6, 12, 16 ], [ 2, 10, 11, 13 ], [ 3, 5, 6, 10 ],
          [ 3, 7, 9, 13 ], [ 3, 11, 12, 14 ], [ 4, 6, 7, 11 ],
          [ 4, 8, 10, 14 ], [ 4, 12, 13, 15 ], [ 5, 7, 8, 12 ],
          [ 5, 13, 14, 16 ], [ 6, 9, 14, 15 ], [ 7, 10, 15, 16 ],
          [ 8, 9, 11, 16 ] ], isBinary := true, isBlockDesign := true,
      v := 16 ) ]
gap> List(d,AllTDesignLambdas);
[ [ 20, 5, 1 ] ]
gap>
```

A GAP session

```
Prune_cs: 792
Prune_only_zeros: 70 of 948
Prune_hoelder: 86 of 878
Prune_N: 0
Fincke-Pohst: 0
Loops: 1756
Total number of solutions: 16

total enumeration time: 0:00:00
Performing isomorph rejection...
1
[ rec( autGroup := <permutation group with 6 generators>, autSubgroup := C8,
      blocks := [ [ 1, 2, 7, 14 ], [ 1, 3, 4, 16 ], [ 1, 5, 11, 15 ],
          [ 1, 6, 8, 13 ], [ 1, 9, 10, 12 ], [ 2, 3, 8, 15 ], [ 2, 4, 5, 9 ],
          [ 2, 6, 12, 16 ], [ 2, 10, 11, 13 ], [ 3, 5, 6, 10 ],
          [ 3, 7, 9, 13 ], [ 3, 11, 12, 14 ], [ 4, 6, 7, 11 ],
          [ 4, 8, 10, 14 ], [ 4, 12, 13, 15 ], [ 5, 7, 8, 12 ],
          [ 5, 13, 14, 16 ], [ 6, 9, 14, 15 ], [ 7, 10, 15, 16 ],
          [ 8, 9, 11, 16 ] ], isBinary := true, isBlockDesign := true,
      v := 16 ) ]
gap> List(d,AllTDesignLambdas);
[ [ 20, 5, 1 ] ]
gap> d:=KramerMesnerSearch(3,16,4,1,g,rec(NonIsomorphic:=true));;
```

A GAP session

```
1000100000000000000010000011000010001000100000001001010011100000000000000001001
1000100000000000000000101111000000101000000000100001110011110000000000000001001
1000100000000000000000101111000010001000100000100001110011110000000000000001001
1000000001000000001000000011000000011000000100001000010011000000010000000001001
1000000001000001000000000111000001001000001000000000110011000000010000000001001
1000010000000000000010000011000000001010000100000010010011001000000000000001001
1000010000000000000010001010000000011000001000000001110011001000000000000001001
1000001000000001000001000001000000001100000000001001010011000010000000000001001
1000001000000001000000010011000000001010000000100010010011000010000000000001001
1000000010000001000010000001000001001000001000000001010011000000000000001001001
1000000010000000001000100010000000110000001001000010100110000000000000001001001
Prune_cs: 2035
Prune_only_zeros: 397 of 3136
Prune_hoelder: 593 of 2739
Prune_N: 121
Fincke-Pohst: 0
Loops: 5323
Total number of solutions: 152

total enumeration time: 0:00:00
Performing isomorph rejection...
30
gap>
```

A GAP session

```
100010000000000000010000011000010001000100000001001010011100000000000000001001
100010000000000000000010111000000101000000000100001110011100000000000000001001
100010000000000000000010111000010001000100000100001110011100000000000000001001
100000000100000000100000001100000001100000010000100001001100000001000000001001
100000000100000100000000011100000100100000100000000011001100000001000000001001
100001000000000000010000011000000001010000100000010010011001000000000000001001
100001000000000000010001010000000011000001000000001110011001000000000000001001
100001000000001000001000001000000001100000000001001010011000010000000000001001
100001000000001000000010011000000001010000000100010010011000010000000000001001
100000001000000100001000000100000100100000100000000010100110000000000001001001
100000001000000000100010001000000011000000100100001010011000000000000001001001
Prune_cs: 2035
Prune_only_zeros: 397 of 3136
Prune_hoelder: 593 of 2739
Prune_N: 121
Fincke-Pohst: 0
Loops: 5323
Total number of solutions: 152

total enumeration time: 0:00:00
Performing isomorph rejection...
30
gap> List(d,AllTDesignLambdas);
```

A GAP session

```
100000001000000100001000000100000100100000100000000101001100000000000000001001001
100000001000000000100001000100000001100000010010000101001100000000000000001001001
Prune_cs: 2035
Prune_only_zeros: 397 of 3136
Prune_hoelder: 593 of 2739
Prune_N: 121
Fincke-Pohst: 0
Loops: 5323
Total number of solutions: 152

total enumeration time: 0:00:00
Performing isomorph rejection...
30
gap> List(d,AllTDesignLambdas);
[ [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ] ]
gap>
```

A GAP session

```
100000001000000100001000000100000100100000100000000101001100000000000001001001
100000001000000000100001000100000001100000010010000101001100000000000001001001
Prune_cs: 2035
Prune_only_zeros: 397 of 3136
Prune_hoelder: 593 of 2739
Prune_N: 121
Fincke-Pohst: 0
Loops: 5323
Total number of solutions: 152

total enumeration time: 0:00:00
Performing isomorph rejection...
30
gap> List(d,AllTDesignLambdas);
[ [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ] ]
gap> c:=(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16);
```

A GAP session

```
Prune_cs: 2035
Prune_only_zeros: 397 of 3136
Prune_hoelder: 593 of 2739
Prune_N: 121
Fincke-Pohst: 0
Loops: 5323
Total number of solutions: 152

total enumeration time: 0:00:00
Performing isomorph rejection...
30
gap> List(d,AllTDesignLambdas);
[ [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ] ]
gap> c:=(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16);
(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16)
gap>
```

A GAP session

```
Prune_cs: 2035
Prune_only_zeros: 397 of 3136
Prune_hoelder: 593 of 2739
Prune_N: 121
Fincke-Pohst: 0
Loops: 5323
Total number of solutions: 152

total enumeration time: 0:00:00
Performing isomorph rejection...
30
gap> List(d,AllTDesignLambdas);
[ [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ] ]
gap> c:=(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16);
(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16)
gap> g3:=Group(a,b,c);
```

A GAP session

```
Prune_N: 121
Fincke-Pohst: 0
Loops: 5323
Total number of solutions: 152

total enumeration time: 0:00:00
Performing isomorph rejection...
30
gap> List(d,AllTDesignLambdas);
[ [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ] ]
gap> c:=(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16);
(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16)
gap> g3:=Group(a,b,c);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16), (1,8)(2,7)(3,6)(4,5)(9,16)
  (10,15)(11,14)(12,13), (1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16) ])
gap>
```

A GAP session

```
Prune_N: 121
Fincke-Pohst: 0
Loops: 5323
Total number of solutions: 152

total enumeration time: 0:00:00
Performing isomorph rejection...
30
gap> List(d,AllTDesignLambdas);
[ [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ] ]
gap> c:=(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16);
(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16)
gap> g3:=Group(a,b,c);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16), (1,8)(2,7)(3,6)(4,5)(9,16)
  (10,15)(11,14)(12,13), (1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16) ])
gap> StructureDescription(g3);
```

A GAP session

```
Loops: 5323
Total number of solutions: 152

total enumeration time: 0:00:00
Performing isomorph rejection...
30
gap> List(d,AllTDesignLambdas);
[ [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ] ]
gap> c:=(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16);
(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16)
gap> g3:=Group(a,b,c);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16), (1,8)(2,7)(3,6)(4,5)(9,16)
  (10,15)(11,14)(12,13), (1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16) ])
gap> StructureDescription(g3);
"C2 x D16"
gap>
```

A GAP session

```
Loops: 5323
Total number of solutions: 152

total enumeration time: 0:00:00
Performing isomorph rejection...
30
gap> List(d,AllTDesignLambdas);
[ [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ],
  [ 140, 35, 7, 1 ], [ 140, 35, 7, 1 ] ]
gap> c:=(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16);
(1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16)
gap> g3:=Group(a,b,c);
Group([ (1,2,3,4,5,6,7,8)(9,10,11,12,13,14,15,16), (1,8)(2,7)(3,6)(4,5)(9,16)
  (10,15)(11,14)(12,13), (1,9)(2,10)(3,11)(4,12)(5,13)(6,14)(7,15)(8,16) ])
gap> StructureDescription(g3);
"C2 x D16"
gap> d:=KramerMesnerSearch(5,16,6,3,g3,rec(NonIsomorphic:=true));;
```

A GAP session

```
1000000 loops, solutions: 4, fipo: 0
2000000 loops, solutions: 4, fipo: 0
3000000 loops, solutions: 4, fipo: 0
4000000 loops, solutions: 4, fipo: 0
5000000 loops, solutions: 4, fipo: 0
6000000 loops, solutions: 4, fipo: 0
7000000 loops, solutions: 4, fipo: 0
8000000 loops, solutions: 4, fipo: 0
9000000 loops, solutions: 4, fipo: 0
10000000 loops, solutions: 4, fipo: 0
11000000 loops, solutions: 4, fipo: 0
10001000001010000011010000100100000000110010100110100100000000001000001011001001010
10001000001010001001010000000101000000101010100110100001000001000010001001001001001
00101001100000100100000000101010001000010110000010000001000001000011001100000001100
00001101100000100101000001000010001010000110000010010000010001001000011000000001100
12000000 loops, solutions: 8, fipo: 0
13000000 loops, solutions: 8, fipo: 0
14000000 loops, solutions: 8, fipo: 0
15000000 loops, solutions: 8, fipo: 0
16000000 loops, solutions: 8, fipo: 0
17000000 loops, solutions: 8, fipo: 0
18000000 loops, solutions: 8, fipo: 0
^CPerforming isomorph rejection...
```

A GAP session

```
3000000 loops, solutions: 4, fipo: 0
4000000 loops, solutions: 4, fipo: 0
5000000 loops, solutions: 4, fipo: 0
6000000 loops, solutions: 4, fipo: 0
7000000 loops, solutions: 4, fipo: 0
8000000 loops, solutions: 4, fipo: 0
9000000 loops, solutions: 4, fipo: 0
10000000 loops, solutions: 4, fipo: 0
11000000 loops, solutions: 4, fipo: 0
1000100000101000001101000010010000000011001010011010010000000000100000101100100101010
1000100000101000100101000000010100000010101010011010000100000100001000100100100100100
0010100110000010010000000010101000100001011000001000000100000100001100110000000110000
0000110110000001001010000010000100010100001100000100100000100010010000110000000011000
12000000 loops, solutions: 8, fipo: 0
13000000 loops, solutions: 8, fipo: 0
14000000 loops, solutions: 8, fipo: 0
15000000 loops, solutions: 8, fipo: 0
16000000 loops, solutions: 8, fipo: 0
17000000 loops, solutions: 8, fipo: 0
18000000 loops, solutions: 8, fipo: 0
^CPerforming isomorph rejection...
4
gap>
```

A GAP session

```
3000000 loops, solutions: 4, fipo: 0
4000000 loops, solutions: 4, fipo: 0
5000000 loops, solutions: 4, fipo: 0
6000000 loops, solutions: 4, fipo: 0
7000000 loops, solutions: 4, fipo: 0
8000000 loops, solutions: 4, fipo: 0
9000000 loops, solutions: 4, fipo: 0
10000000 loops, solutions: 4, fipo: 0
11000000 loops, solutions: 4, fipo: 0
100010000010100000110100001001000000001100101001101001000000000010000010110010010101
100010000010100010010100000001010000001010101001101000010000010000100010010010010011
001010011000001001000000001010100010000101100000100000010000010000110011000000011000
000011011000001001010000010000100010100001100000100100000100010010000110000000011000
12000000 loops, solutions: 8, fipo: 0
13000000 loops, solutions: 8, fipo: 0
14000000 loops, solutions: 8, fipo: 0
15000000 loops, solutions: 8, fipo: 0
16000000 loops, solutions: 8, fipo: 0
17000000 loops, solutions: 8, fipo: 0
18000000 loops, solutions: 8, fipo: 0
^CPerforming isomorph rejection...
4
gap> List(d,AllTDesignLambdas);
```

A GAP session

```
6000000 loops, solutions: 4, fipo: 0
7000000 loops, solutions: 4, fipo: 0
8000000 loops, solutions: 4, fipo: 0
9000000 loops, solutions: 4, fipo: 0
10000000 loops, solutions: 4, fipo: 0
11000000 loops, solutions: 4, fipo: 0
10001000001010000011010000100100000000110010100110100100000000001000001011001001010
10001000001010001001010000000101000000101010100110100001000001000010001001001001001
00101001100000100100000000101010001000010110000010000001000001000011001100000001100
00001101100000100101000001000010001010000110000010010000010001001000011000000001100
12000000 loops, solutions: 8, fipo: 0
13000000 loops, solutions: 8, fipo: 0
14000000 loops, solutions: 8, fipo: 0
15000000 loops, solutions: 8, fipo: 0
16000000 loops, solutions: 8, fipo: 0
17000000 loops, solutions: 8, fipo: 0
18000000 loops, solutions: 8, fipo: 0
^CPerforming isomorph rejection...
4
gap> List(d,AllTDesignLambdas);
[ [ 2184, 819, 273, 78, 18, 3 ], [ 2184, 819, 273, 78, 18, 3 ],
  [ 2184, 819, 273, 78, 18, 3 ], [ 2184, 819, 273, 78, 18, 3 ] ]
gap>
```

KramerMesnerSearch: behind the scenes

`KramerMesnerSearch`: behind the scenes

- Generate G-orbits of t-subsets and k-subsets of V [GAP code]

KramerMesnerSearch: behind the scenes

- Generate G-orbits of t-subsets and k-subsets of V [GAP code]
  - $\rightsquigarrow$ Orderly algorithm using GAP package **images**
  - $\rightsquigarrow$ Algorithm for short orbits

KramerMesnerSearch: behind the scenes

- Generate G-orbits of t-subsets and k-subsets of V [GAP code]
  - $\rightsquigarrow$ Orderly algorithm using GAP package **images**
  - $\rightsquigarrow$ Algorithm for short orbits
- Compute the Kramer-Mesner matrix [GAP code]

`KramerMesnerSearch`: behind the scenes

- Generate G-orbits of t-subsets and k-subsets of V [GAP code]
  $\rightsquigarrow$ Orderly algorithm using GAP package **images**
  $\rightsquigarrow$ Algorithm for short orbits
- Compute the Kramer-Mesner matrix [GAP code]
- Solve 0-1 system using A. Wassermann's program `solvediophant` or our own backtracking solver [interface to C programs]

`KramerMesnerSearch`: behind the scenes

- Generate G-orbits of t-subsets and k-subsets of V [GAP code]
  - $\rightsquigarrow$ Orderly algorithm using GAP package **images**
  - $\rightsquigarrow$ Algorithm for short orbits
- Compute the Kramer-Mesner matrix [GAP code]
- Solve 0-1 system using A. Wassermann's program `solvediophant` or our own backtracking solver [interface to C programs]
- Transform solutions to **Design** package format [GAP code]

`KramerMesnerSearch`: behind the scenes

- Generate G-orbits of t-subsets and k-subsets of V [GAP code]
  $\rightsquigarrow$ Orderly algorithm using GAP package **images**
  $\rightsquigarrow$ Algorithm for short orbits
- Compute the Kramer-Mesner matrix [GAP code]
- Solve 0-1 system using A. Wassermann's program `solvediophant` or our own backtracking solver [interface to C programs]
- Transform solutions to **Design** package format [GAP code]
- Isomorph rejection using B. D. McKay's program `nauty` [interface to C program]

PAG – Implemented so far

Version 0.1.2:

- Auxiliary functions for permutation groups: `MoveGroup`, `MultiGroup`, `RestrictedGroup`, `AllSubgroupsConjugation...`

PAG – Implemented so far

Version 0.1.2:

- Auxiliary functions for permutation groups: `MoveGroup`, `MultiGroup`, `RestrictedGroup`, `AllSubgroupsConjugation...`
- Functions for generating orbit representatives: `SubsetOrbitRep`, `SmallLambdaFilter`, `IsGoodSubsetOrbit...`

PAG – Implemented so far

Version 0.1.2:

- Auxiliary functions for permutation groups: `MoveGroup`, `MultiGroup`, `RestrictedGroup`, `AllSubgroupsConjugation...`
- Functions for generating orbit representatives: `SubsetOrbitRep`, `SmallLambdaFilter`, `IsGoodSubsetOrbit...`
- Support functions for Kramer-Mesner method: `KramerMesnerMat`, `CompatibilityMat`, `SolveKramerMesner`, `BaseBlocks...`

PAG – Implemented so far

Version 0.1.2:

- Auxiliary functions for permutation groups: `MoveGroup`, `MultiGroup`, `RestrictedGroup`, `AllSubgroupsConjugation...`
- Functions for generating orbit representatives: `SubsetOrbitRep`, `SmallLambdaFilter`, `IsGoodSubsetOrbit...`
- Support functions for Kramer-Mesner method: `KramerMesnerMat`, `CompatibilityMat`, `SolveKramerMesner`, `BaseBlocks...`
- Inspecting objects: `IntersectionNumbers`, `BlockScheme...`

PAG – Implemented so far

Version 0.1.2:

- Auxiliary functions for permutation groups: `MoveGroup`, `MultiGroup`, `RestrictedGroup`, `AllSubgroupsConjugation...`
- Functions for generating orbit representatives: `SubsetOrbitRep`, `SmallLambdaFilter`, `IsGoodSubsetOrbit...`
- Support functions for Kramer-Mesner method: `KramerMesnerMat`, `CompatibilityMat`, `SolveKramerMesner`, `BaseBlocks...`
- Inspecting objects: `IntersectionNumbers`, `BlockScheme...`

Version 0.1.4:

- Latin squares and MOLS sets with prescribed autotopism groups: `KramerMesnerMOLS`, `MOLSAut`, `MOLSFilter...`

PAG – Future developments

Enhancements of the Kramer-Mesner method:

PAG – Future developments

Enhancements of the Kramer-Mesner method:

- Tactical decomposition matrices

PAG – Future developments

Enhancements of the Kramer-Mesner method:

- Tactical decomposition matrices
- Quasi-symmetric designs: good orbits, compatibility matrices

PAG – Future developments

Enhancements of the Kramer-Mesner method:

- Tactical decomposition matrices
- Quasi-symmetric designs: good orbits, compatibility matrices
- More solvers: Gurobi, Minion...

PAG – Future developments

Enhancements of the Kramer-Mesner method:

- Tactical decomposition matrices
- Quasi-symmetric designs: good orbits, compatibility matrices
- More solvers: Gurobi, Minion...

Other construction methods and types of objects:

PAG – Future developments

Enhancements of the Kramer-Mesner method:

- Tactical decomposition matrices
- Quasi-symmetric designs: good orbits, compatibility matrices
- More solvers: Gurobi, Minion...

Other construction methods and types of objects:

- Searching for objects by clique search

PAG – Future developments

Enhancements of the Kramer-Mesner method:

- Tactical decomposition matrices
- Quasi-symmetric designs: good orbits, compatibility matrices
- More solvers: Gurobi, Minion...

Other construction methods and types of objects:

- Searching for objects by clique search
- Configurations

PAG – Future developments

Enhancements of the Kramer-Mesner method:

- Tactical decomposition matrices
- Quasi-symmetric designs: good orbits, compatibility matrices
- More solvers: Gurobi, Minion...

Other construction methods and types of objects:

- Searching for objects by clique search
- Configurations
- Strongly regular graphs

The End

Thanks for your attention!