`group.m`

Supported groups are direct products of finite groups and some Lie groups. Supported finite groups are those whose irreps was calculated by `GAP` (in `sgd/`), and any dihedral and quartenion groups. Supported Lie groups are `su[2]`, `so[2]`, `o[2]`, `so[3]`, `o[3]`. We support only compact groups, so we can assume any finite dimensional irrep can be unitarized.

This package imports `groupd.m` and `grouplie.m`.

- Get Groups
- Group Data
- Irrep-Objects
- `groupd.m`
- `grouplie.m`

Get Groups

`getGroup`

`getGroup[g,i]` loads data from `sgd/sg.g.i.m` and returns group-object `group[g,i]`. `g` is the order of the finite group, `i` is the number of the group assined by `GAP`.

`product`

`product[g1,g2]` returns group-object `pGroup[g1,g2]` which represents direct product of two group-object `g1`, `g2`.

`group`

`group[g,i]` is a group-object whose order is `g` and whose number assigned by `GAP` is `i`. Before using this value, you have to call `getGroup[g,i]` to get proper group-object.

`pGroup`

`pGroup[g1,g2]` is a group-object which is a direct product of `g1`, `g2`. Before using this value, you have to call `product[g1,g2]` to get proper group-object.

`setGroup`

`setGroup[G]` loads `inv.m` with global symmetry `G`. This action clears all values calculated by `inv.m` previously.

`available`

`available[g,i]` gives whether `group[g,i]` are supported or not.

Group Data

A group-object `g` has attributes `ncg`, `ct`, `id`, `dim`, `prod`, `dual`, `isrep`, `gG`, `gA`, `minrep`. You can evaluate attributes in putting it in `g[...]`. For example, `g[dim[r]]` gives the dimension of irrep `r`.

`ncg`

`ncg` is the number of conjugacy classes, which is also the number of inequivalent irreps. This is not defined for Lie groups.

`ct`

`ct` is the character table. This is not defined for Lie groups.

`id`

`id` is the trivial representation.

`dim`

`dim[r]` is the dimension of irrep `r`.

`prod`

`prod[r,s]` gives a list of all irreps arising in irreducible decomposition of direct product representation of `r` and `s`. `prod[r,s]` may not be duplicate-free.

`dual`

`dual[r]` gives dual representation of irrep `r`.

`isrep`

`isrep[r]` gives whether `r` is recognised as a irrep-object of the group-object or not.

`gG`

`gG` is a list of all generator-objects of finite group part of the group-object.

`gA`

`gA` is a list of all generator-objects of Lie algebra part of the group-object.

`minrep`

`minrep[r,s]` gives `r` if `r < s` else `s`. `r` and `s` are irrep-objects.

Irrep-Objects

We need all irreps to be sorted in some linear order. All irrep-objects of `G=group[g,i]` are `rep[1]`, `rep[2]`, ..., `rep[n]` (`n=G[ncg]`). all irrep-objects of `pGroup[g1,g2]` are `rep[r1,r2]`, where `r1` is a irrep-object of `g1` and `r2` is a irrep-object of `g2`. `minrep` compares irreps in lexical order.

`rep`

`rep[n]` is `n`-th irrep-object (`n` is assined by `GAP` and corresponds to the index of `ct`). This is recognised only by `group[g,i]`.

`rep[r1,r2]` is natural irrep-object of `pGroup[g1,g2]` where `r1` is irrep-object of `g1`, `r2` is irrep-object of `g2`. This is recognised only by `pGroup[g1,g2]`.

`v`

`v[n]` is spin-`n` irrep-object. This is recognised only by `dih[n]`, `dic[n]`, `su[2]`, `so[3]`, `o[2]` and `so[2]`.

`v[n,s]` is spin-`n` irrep-object with sign `s`. This is recognised only by `o[3]`.

`i`

`i[a]` is one-dimensional irrep-object with sign `a`. This is recognised only by `dih[n]` (`n`: odd) and `o[2]`.

`i[a,b]` is one-dimensional irrep-object with sign `a,b`. This is recognised only by `dih[n]` (`n`:even), `dic[n]`.

`groupd.m`

If `n` is even, all irrep-objects of `dih[n]` are `i[1,1], i[1,-1], i[-1,1], i[-1,-1], v[1], ..., v[n/2-1]`. If `n` is odd, all irrep-objects of `dih[n]` are `i[1], i[-1] v[1], ..., v[(n-1)/2]`. All irrep-objects of `dic[n]` are `i[1,1], i[1,-1], i[-1,1], i[-1,-1], v[1], ..., v[n-1]`.

`getDihedral`

`getDihedral[n]` returns group-object `dih[n]` which represents the dihedral group of order `2n`.

`getDicyclic`

`getDicyclic[n]` returns group-object `dic[n]` which represents the dicyclic group of order `4n`.

`dih`

`dih[n]` is a group-object which is the dihedral group of order `2n`. Before using this value, you have to call `getDihedral[n]` to get proper group-object.

`dic`

`dic[n]` is a group-object which is the dicyclic group of order `4n`. Before using this value, you have to call `getDicyclic[n]` to get proper group-object.

`grouplie.m`

All irrep-objects of `G=su[2]` are `v[0], v[1/2], v[1], v[3/2], ...`. All irrep-objects of `G=o[3]` are `v[0,1], v[0,-1], v[1,1], v[1,-1], v[2,1], v[2,-1], v[3,1], v[3,-1], ...`. All irrep-objects of `G=so[3]` are `v[0], v[1], v[2], v[3], ...`. All irrep-objects of `G=o[2]` are `i[1], i[-1], v[1], v[2], v[3], ...`. All irrep-objects of `G=so[2]` are `v[x]` (`x \in \mathbb{R}`).

`getSU`

`getSU[n]` returns group-object `su[n]` which represents the special unitary group of rank `n`. `n` must be `2,4`.

`getO`

`getO[n]` returns group-object `o[n]` which represents the orthogonal group of rank `n`. `n` must be `2,3`.

`getSO`

`getSO[n]` returns group-object `su[n]` which represents the special orthogonal group of rank `n`. `n` must be `2,3`.

`su`

`su[n]` is a group-object which is the special unitary group of rank `n`. Before using this value, you have to call `getSU[n]` to get proper group-object.

`o`

`o[n]` is a group-object which is the orthogonal group of rank `n`. Before using this value, you have to call `getO[n]` to get proper group-object.

`so`

`so[n]` is a group-object which is the special orthogonal group of rank `n`. Before using this value, you have to call `getSO[n]` to get proper group-object.