Permuta Documentation

Release 0.1

Úlfarsson

Jun 20, 2017

Contents

Permuta is a Python library for working with permutations and mesh patterns. Table of contents:

CHAPTER 1

Installing

To install Permuta on your system, simply run the following command as a superuser:

```
./setup.py install
```

It is also possible to install Permuta in development mode, in which case you run the following instead:

```
./setup.py develop
```

To run the unit tests, you can run the following command:

```
./setup.py test
```

Once you've installed Permuta, it can be imported into a Python script just like any other Python library:

```
from permuta import Permutation, MeshPattern
```

CHAPTER 2

License

BSD-3

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

CHAPTER 3

permuta package

Subpackages

permuta.math package

Submodules

permuta.math.counting module

permuta.misc package

Submodules

permuta.misc.algorithm_x module

permuta.misc.dancing_links module

permuta.misc.exact_cover module

permuta.misc.misc module

permuta.misc.ordered_set_partitions module

permuta.misc.progressbar module

permuta.misc.triemap module

permuta.misc.union_find module

class permuta.misc.union_find.**UnionFind**(*n=0*)

Bases: object

A collection of distjoint sets.

add()
Add a unit set containing a new element to the collection, and return the identifier of the new element.

find(*x*)
Return the identifier of a representative element for the set containing the element with identifier x.

size(*x*)
Return the number of elements in the set containing the element with identifier x.

unite(*x*, *y*)
Unite the two sets containing the elements with identifiers x and y, respectively.

Submodules

permuta.mesh_pattern module

permuta.mesh_patterns module

permuta.permutation module

permuta.permutations module

CHAPTER 4

Indices and tables

- genindex
- modindex
- search

Python Module Index

p

Index