

Python Webgraph Generator

Generated by Doxygen 1.5.6

Sat Dec 13 02:15:44 2008

Contents

1	Main Page	1
1.1	Description	1
1.2	Todo Lists	1
2	General Todo List	3
3	Todo List	5
4	Module Index	7
4.1	Modules	7
5	Namespace Index	9
5.1	Namespace List	9
6	Class Index	11
6.1	Class Hierarchy	11
7	Class Index	13
7.1	Class List	13
8	Module Documentation	15
8.1	Basic Elements	15
8.2	Package Exceptions	16
8.3	Graph	17
8.4	Random Graphs	18
8.5	System	19
9	Namespace Documentation	21
9.1	BaseElements Namespace Reference	21
9.1.1	Detailed Description	21
9.2	Exceptions Namespace Reference	22

9.2.1	Detailed Description	22
9.3	Graph Namespace Reference	23
9.3.1	Detailed Description	23
9.4	PackageExceptions Namespace Reference	24
9.4.1	Detailed Description	24
9.5	RandomGraphs Namespace Reference	25
9.5.1	Detailed Description	25
9.6	System Namespace Reference	26
9.6.1	Detailed Description	26
10	Class Documentation	27
10.1	pygel::BaseElements::AbstractEdge::AbstractEdge Class Reference	27
10.1.1	Detailed Description	27
10.1.2	Member Data Documentation	28
10.1.2.1	getStartVertex	28
10.1.2.2	getEndVertex	28
10.1.2.3	setStartVertex	28
10.1.2.4	setEndVertex	28
10.2	pygel::Graph::AbstractGraph::AbstractGraph Class Reference	29
10.2.1	Detailed Description	29
10.3	pygel::BaseElements::AbstractVertex::AbstractVertex Class Reference	30
10.3.1	Detailed Description	30
10.4	pygel::RandomGraphs::ChooseEdges::ChooseEdges Class Reference	31
10.4.1	Detailed Description	31
10.4.2	Member Function Documentation	32
10.4.2.1	__init__	32
10.4.2.2	selectVertex	32
10.4.3	Member Data Documentation	32
10.4.3.1	serialEdgeList	32
10.4.3.2	startVertX	32
10.4.3.3	endVertX	32
10.4.3.4	startVertY	33
10.4.3.5	endVertY	33
10.4.3.6	noOfEdges	33
10.4.3.7	probA	33
10.4.3.8	probB	33
10.4.3.9	probC	33

10.4.3.10	probD	33
10.4.3.11	noSelfLoops	33
10.5	pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph Class Reference	34
10.5.1	Detailed Description	34
10.5.2	Member Function Documentation	35
10.5.2.1	__init__	35
10.5.2.2	setProbs	35
10.5.2.3	generate	36
10.5.2.4	populate	36
10.5.2.5	writeEdges	36
10.5.3	Member Data Documentation	36
10.5.3.1	probA	36
10.5.3.2	serialEdgeList	37
10.6	pygel::Exceptions::Exceptions::DistError Class Reference	38
10.6.1	Detailed Description	38
10.6.2	Member Function Documentation	38
10.6.2.1	__init__	38
10.7	pygel::BaseElements::Edge::Edge Class Reference	39
10.7.1	Detailed Description	39
10.7.2	Member Function Documentation	40
10.7.2.1	__init__	40
10.7.2.2	getStartVertex	40
10.7.2.3	getEndVertex	40
10.7.2.4	setStartVertex	40
10.7.2.5	setEndVertex	40
10.8	pygel::Exceptions::Exceptions::EdgeError Class Reference	41
10.8.1	Detailed Description	41
10.8.2	Member Function Documentation	41
10.8.2.1	__init__	41
10.8.2.2	__init__	42
10.8.2.3	__init__	42
10.9	pygel::Exceptions::Exceptions::Error Class Reference	43
10.9.1	Detailed Description	43
10.10	pygel::Exceptions::Exceptions::ErrorMessages Class Reference	44
10.10.1	Detailed Description	44

10.11 <code>pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph</code> Class Reference	45
10.11.1 Detailed Description	45
10.11.2 Member Function Documentation	47
10.11.2.1 <code>addEdge</code>	47
10.11.2.2 <code>deleteEdge</code>	47
10.11.2.3 <code>addVertex</code>	48
10.11.2.4 <code>deleteVertex</code>	48
10.11.2.5 <code>getEdges</code>	48
10.11.2.6 <code>getVertices</code>	48
10.11.2.7 <code>getLastEdgeNumber</code>	49
10.11.2.8 <code>getOutNeighbors</code>	49
10.11.2.9 <code>getInNeighbors</code>	49
10.11.2.10 <code>getNumberOfOutNeighbors</code>	49
10.11.2.11 <code>getNumberOfInNeighbors</code>	50
10.11.2.12 <code>getNumberOfNeighbors</code>	50
10.11.2.13 <code>getInDegreeDistribution</code>	50
10.11.2.14 <code>getOutDegreeDistribution</code>	50
10.11.2.15 <code>getJointDistribution</code>	51
10.11.2.16 <code>getDegreeDistribution</code>	51
10.11.2.17 <code>getVerticesByInDegree</code>	51
10.11.2.18 <code>getVerticesByOutDegree</code>	51
10.11.2.19 <code>getSCComponents</code>	52
10.11.2.20 <code>getOutComponent</code>	52
10.11.2.21 <code>writeCC</code>	52
10.11.2.22 <code>writeEdges</code>	52
10.11.2.23 <code>readEdges</code>	53
10.11.2.24 <code>findEdge</code>	53
10.11.2.25 <code>findVertex</code>	53
10.11.2.26 <code>hasVertex</code>	54
10.12 <code>pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph</code> Class Reference	55
10.12.1 Detailed Description	55
10.12.2 Member Function Documentation	57
10.12.2.1 <code>addEdge</code>	57
10.12.2.2 <code>deleteEdge</code>	57
10.12.2.3 <code>addVertex</code>	57

10.12.2.4	deleteVertex	57
10.12.2.5	getEdges	58
10.12.2.6	getVertices	58
10.12.2.7	getVertexNumbers	58
10.12.2.8	getLastEdgeNumber	58
10.12.2.9	getNeighbors	58
10.12.2.10	getNumberOfNeighbors	59
10.12.2.11	getDegreeDistribution	59
10.12.2.12	getSCComponents	59
10.12.2.13	writeCC	59
10.12.2.14	writeEdges	60
10.12.2.15	readEdges	60
10.12.2.16	findEdge	60
10.12.2.17	findVertex	60
10.12.2.18	hasVertex	61
10.13	pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph	
	Class Reference	62
10.13.1	Detailed Description	62
10.13.2	Member Function Documentation	63
10.13.2.1	__init__	63
10.13.2.2	setProbs	63
10.13.2.3	generate	64
10.13.2.4	populate	64
10.13.2.5	writeEdges	64
10.13.3	Member Data Documentation	64
10.13.3.1	probA	64
10.13.3.2	serialEdgeList	65
10.14	pygel::BaseElements::Vertex::Vertex Class Reference	66
10.14.1	Detailed Description	66
10.14.2	Member Function Documentation	66
10.14.2.1	__init__	66
10.14.2.2	getVertexNumber	66
10.14.2.3	setVertexNumber	67
10.15	pygel::Exceptions::Exceptions::VertexError Class Reference	68
10.15.1	Detailed Description	68
10.15.2	Member Function Documentation	68
10.15.2.1	__init__	68

10.16	pygel::BaseElements::WeightedVertex::WeightedVertex Class Reference	69
10.16.1	Detailed Description	69
10.16.2	Member Function Documentation	69
10.16.2.1	__init__	69
10.16.2.2	getWeight	69
10.16.2.3	setWeight	70
10.17	pygel::BaseElements::WeightedVertices::WeightedVertices Class Reference	71
10.17.1	Detailed Description	71
10.17.2	Member Function Documentation	71
10.17.2.1	__init__	71
10.17.2.2	addVertex	71
10.17.2.3	delVertex	72
10.17.2.4	getVertices	72
10.17.2.5	findVertex	72
10.17.2.6	findWeight	72
10.17.2.7	hasVertex	73

Chapter 1

Main Page

1.1 Description

A threaded Web graph (Power law random graph) generator written in Python. It can generate a synthetic Web graph of about one million nodes in a few minutes on a desktop machine. It implements a threaded variant of the RMAT algorithm. A little tweak can produce graphs representing social-networks or community-networks

1.2 Todo Lists

- [General Todo List](#)
- [Todo List](#)

Chapter 2

General Todo List

Todo

Integrate with [psyco](#)

Chapter 3

Todo List

Member [pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph::probA](#)
Add description about choosing these probabilities

Member [pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph::probA](#)
Add description about choosing these probabilities

Page [General Todo List](#) Integrate with [psyco](#)

Chapter 4

Module Index

4.1 Modules

Here is a list of all modules:

Basic Elements	15
Package Exceptions	16
Graph	17
Random Graphs	18
System	19

Chapter 5

Namespace Index

5.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

BaseElements (Module for defining basic elements)	21
Exceptions (Module for handling exceptions)	22
Graph (Module for handling graphs)	23
PackageExceptions (A module handling package exceptions)	24
RandomGraphs (Module for handling random graphs)	25
System (Module for handling common system related things (for ex)	26

Chapter 6

Class Index

6.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

pygel::BaseElements::AbstractEdge::AbstractEdge	27
pygel::BaseElements::Edge::Edge	39
pygel::Graph::AbstractGraph::AbstractGraph	29
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph	45
pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph	34
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph	55
pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph	62
pygel::BaseElements::AbstractVertex::AbstractVertex	30
pygel::BaseElements::Vertex::Vertex	66
pygel::RandomGraphs::ChooseEdges::ChooseEdges	31
pygel::Exceptions::Exceptions::Error	43
pygel::Exceptions::Exceptions::DistError	38
pygel::Exceptions::Exceptions::EdgeError	41
pygel::Exceptions::Exceptions::VertexError	68
pygel::Exceptions::Exceptions::ErrorMessage	44
pygel::BaseElements::WeightedVertex::WeightedVertex	69
pygel::BaseElements::WeightedVertices::WeightedVertices	71

Chapter 7

Class Index

7.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

pygel::BaseElements::AbstractEdge::AbstractEdge (Abstract class for representing an edge) . . .	27
pygel::Graph::AbstractGraph::AbstractGraph (Abstract class for representing a graph)	29
pygel::BaseElements::AbstractVertex::AbstractVertex (Abstract class for representing a vertex)	30
pygel::RandomGraphs::ChooseEdges::ChooseEdges (Thread for selecting a set of edges)	31
pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph (Generates a synthetic Web graph or Power Law graph using an RMAT algorithm) . .	34
pygel::Exceptions::Exceptions::DistError (Represents a DistError exception)	38
pygel::BaseElements::Edge::Edge (Represents graph edge)	39
pygel::Exceptions::Exceptions::EdgeError (Represents a EdgeError exception)	41
pygel::Exceptions::Exceptions::Error (Empty base class from which all exceptions are derived)	43
pygel::Exceptions::Exceptions::ErrorMessage (Collection of various error message strings) . .	44
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph (Represents a numbered edge graph)	45
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph (Represents a numbered edge graph)	55
pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph (Generates a synthetic Web graph or Power Law graph using an RMAT algorithm) . .	62
pygel::BaseElements::Vertex::Vertex (Represents graph vertex)	66
pygel::Exceptions::Exceptions::VertexError (Represents a VertexError exception)	68
pygel::BaseElements::WeightedVertex::WeightedVertex (Represents a weighted vertex)	69
pygel::BaseElements::WeightedVertices::WeightedVertices (Represents a collection of weighted vertices of type BaseElements::WeightedVertices)	71

Chapter 8

Module Documentation

8.1 Basic Elements

Classes

- class `pygel::BaseElements::AbstractEdge::AbstractEdge`
Abstract class for representing an edge.
- class `pygel::BaseElements::AbstractVertex::AbstractVertex`
Abstract class for representing a vertex.
- class `pygel::BaseElements::Edge::Edge`
Represents graph edge.
- class `pygel::BaseElements::Vertex::Vertex`
Represents graph vertex.
- class `pygel::BaseElements::WeightedVertex::WeightedVertex`
Represents a weighted vertex.
- class `pygel::BaseElements::WeightedVertices::WeightedVertices`
Represents a collection of weighted vertices of type `BaseElements::WeightedVertices`.

8.2 Package Exceptions

Classes

- class `pygel::Exceptions::Exceptions::Error`
Empty base class from which all exceptions are derived.
- class `pygel::Exceptions::Exceptions::VertexError`
Represents a `VertexError` exception.
- class `pygel::Exceptions::Exceptions::EdgeError`
Represents a `EdgeError` exception.
- class `pygel::Exceptions::Exceptions::DistError`
Represents a `DistError` exception.
- class `pygel::Exceptions::Exceptions::ErrorMessages`
Collection of various error message strings.

8.3 Graph

Classes

- class `pygel::Graph::AbstractGraph::AbstractGraph`
Abstract class for representing a graph.
- class `pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph`
Represents a numbered edge graph.
- class `pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph`
Represents a numbered edge graph.

8.4 Random Graphs

Classes

- class [pygel::RandomGraphs::ChooseEdges::ChooseEdges](#)
Thread for selecting a set of edges.
- class [pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph](#)
Generates a synthetic Web graph or Power Law graph using an RMAT algorithm.
- class [pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph](#)
Generates a synthetic Web graph or Power Law graph using an RMAT algorithm.

8.5 System

Chapter 9

Namespace Documentation

9.1 BaseElements Namespace Reference

9.1.1 Detailed Description

Module for defining basic elements.

9.2 Exceptions Namespace Reference

9.2.1 Detailed Description

Module for handling exceptions.

9.3 Graph Namespace Reference

9.3.1 Detailed Description

Module for handling graphs.

9.4 PackageExceptions Namespace Reference

9.4.1 Detailed Description

A module handling package exceptions.

9.5 RandomGraphs Namespace Reference

9.5.1 Detailed Description

Module for handling random graphs.

9.6 System Namespace Reference

9.6.1 Detailed Description

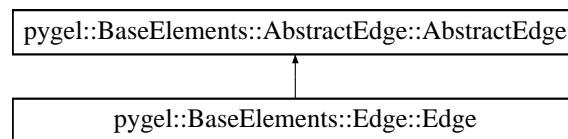
Module for handling common system related things (for ex. logging)

Chapter 10

Class Documentation

10.1 pygel::BaseElements::AbstractEdge::AbstractEdge Class Reference

Inheritance diagram for pygel::BaseElements::AbstractEdge::AbstractEdge::



10.1.1 Detailed Description

Abstract class for representing an edge.

Static Public Attributes

- tuple `getStartVertex` = `AbstractMethod('getStartVertex')`
Abstract method for obtaining the start vertex of an edge.
- tuple `getEndVertex` = `AbstractMethod('getEndVertex')`
Abstract method for obtaining the end vertex of an edge.
- tuple `setStartVertex` = `AbstractMethod('getStartVertex')`
Abstract method for setting the start vertex of an edge.
- tuple `setEndVertex` = `AbstractMethod('getEndVertex')`
Abstract method for setting the end vertex of an edge.

10.1.2 Member Data Documentation

10.1.2.1 `tuple pygel::BaseElements::AbstractEdge::AbstractEdge::getStartVertex = AbstractMethod('getStartVertex') [static]`

Abstract method for obtaining the start vertex of an edge.

10.1.2.2 `tuple pygel::BaseElements::AbstractEdge::AbstractEdge::getEndVertex = AbstractMethod('getEndVertex') [static]`

Abstract method for obtaining the end vertex of an edge.

10.1.2.3 `tuple pygel::BaseElements::AbstractEdge::AbstractEdge::setStartVertex = AbstractMethod('getStartVertex') [static]`

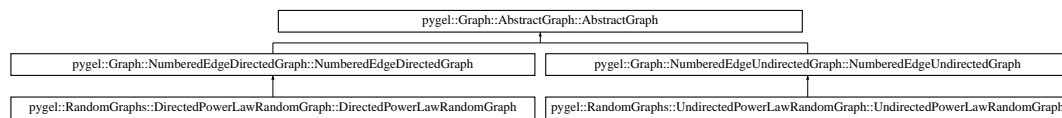
Abstract method for setting the start vertex of an edge.

10.1.2.4 `tuple pygel::BaseElements::AbstractEdge::AbstractEdge::setEndVertex = AbstractMethod('getEndVertex') [static]`

Abstract method for setting the end vertex of an edge.

10.2 pygel::Graph::AbstractGraph::AbstractGraph Class Reference

Inheritance diagram for pygel::Graph::AbstractGraph::AbstractGraph::



10.2.1 Detailed Description

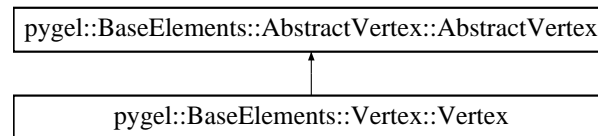
Abstract class for representing a graph.

Static Public Attributes

- tuple `addEdge` = `AbstractMethod('addEdge')`
Abstract method for adding an edge.
- tuple `deleteEdge` = `AbstractMethod('deleteEdge')`
Abstract method for deleting an edge.
- tuple `addVertex` = `AbstractMethod('addVertex')`
Abstract method for adding a vertex.
- tuple `deleteVertex` = `AbstractMethod('deleteVertex')`
Abstract method for deleting a vertex.
- tuple `getEdges` = `AbstractMethod('getEdges')`
Abstract method for obtaining all edges.
- tuple `getVertices` = `AbstractMethod('getVertices')`
Abstract method for obtaining all vertices.

10.3 pygel::BaseElements::AbstractVertex::AbstractVertex Class Reference

Inheritance diagram for pygel::BaseElements::AbstractVertex::AbstractVertex::



10.3.1 Detailed Description

Abstract class for representing a vertex.

Static Public Attributes

- tuple `getVertexNumber` = `AbstractMethod('getVertexNumber')`
Abstract method for obtaining the vertex number.
- tuple `setVertexNumber` = `AbstractMethod('setVertexNumber')`
Abstract method for setting the vertex number.

10.4 pygel::RandomGraphs::ChooseEdges::ChooseEdges Class Reference

10.4.1 Detailed Description

Thread for selecting a set of edges.

Public Member Functions

- def [__init__](#)
Constructs a selector thread.
- def [selectVertex](#)
Selects start and end vertices recursively.
- def [run](#)
Start the thread.

Public Attributes

- [startVertX](#)
- [endVertX](#)
- [startVertY](#)
- [endVertY](#)
- [noOfEdges](#)
- [probA](#)
- [probB](#)
- [probC](#)
- [probD](#)
- [debug](#)
- [noSelfLoops](#)
- [id](#)
Thread ID.

Static Public Attributes

- list [serialEdgeList](#) = []
Common serial edge list.
- tuple [lck](#) = threading.Lock()
Lock that a thread acquires for performing a semaphoric operation.
- tuple [evnt](#) = threading.Event()
- int [id](#) = 0
Thread ID.

10.4.2 Member Function Documentation

10.4.2.1 `def pygel::RandomGraphs::ChooseEdges::ChooseEdges::__init__ (self, noOfEdges, noSelfLoops, startVertX, endVertX, startVertY, endVertY, probA, probB, probC, probD)`

Constructs a selector thread.

See also:

RandomGraphs::DirectedPowerLawRandomGraph

10.4.2.2 `def pygel::RandomGraphs::ChooseEdges::ChooseEdges::selectVertex (self, sVertX, eVertX, sVertY, eVertY, cumulativeA, cumulativeB, cumulativeC)`

Selects start and end vertices recursively.

Parameters:

sVertX Starting column of the adjacency matrix
eVertX Ending column of the adjacency matrix
sVertY Starting row of the adjacency matrix
eVertY Ending column of the adjacency matrix
cumulativeA Cumulative distribution
cumulativeB Cumulative distribution
cumulativeC Cumulative distribution

Returns:

Selected vertices

10.4.3 Member Data Documentation

10.4.3.1 `list pygel::RandomGraphs::ChooseEdges::ChooseEdges::serialEdgeList = []` `[static]`

Common serial edge list.

Updated by each thread in a semaphoric operation

10.4.3.2 `pygel::RandomGraphs::ChooseEdges::ChooseEdges::startVertX`

See also:

RandomGraphs::DirectedPowerLawRandomGraph

10.4.3.3 `pygel::RandomGraphs::ChooseEdges::ChooseEdges::endVertX`

See also:

RandomGraphs::DirectedPowerLawRandomGraph

10.4.3.4 pygel::RandomGraphs::ChooseEdges::ChooseEdges::startVertY

See also:

RandomGraphs::DirectedPowerLawRandomGraph

10.4.3.5 pygel::RandomGraphs::ChooseEdges::ChooseEdges::endVertY

See also:

RandomGraphs::DirectedPowerLawRandomGraph

10.4.3.6 pygel::RandomGraphs::ChooseEdges::ChooseEdges::noOfEdges

See also:

RandomGraphs::DirectedPowerLawRandomGraph

10.4.3.7 pygel::RandomGraphs::ChooseEdges::ChooseEdges::probA

See also:

RandomGraphs::DirectedPowerLawRandomGraph

10.4.3.8 pygel::RandomGraphs::ChooseEdges::ChooseEdges::probB

See also:

RandomGraphs::DirectedPowerLawRandomGraph

10.4.3.9 pygel::RandomGraphs::ChooseEdges::ChooseEdges::probC

See also:

RandomGraphs::DirectedPowerLawRandomGraph

10.4.3.10 pygel::RandomGraphs::ChooseEdges::ChooseEdges::probD

See also:

RandomGraphs::DirectedPowerLawRandomGraph

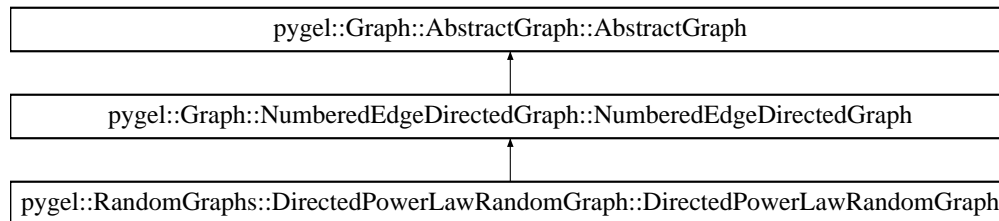
10.4.3.11 pygel::RandomGraphs::ChooseEdges::ChooseEdges::noSelfLoops

See also:

RandomGraphs::DirectedPowerLawRandomGraph

10.5 pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph Class Reference

Inheritance diagram for pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph::



10.5.1 Detailed Description

Generates a synthetic Web graph or Power Law graph using an RMat algorithm.

Public Member Functions

- def `__init__`
Constructs an empty graph.
- def `setProbs`
Sets the probability with which quadrants in an adjacency matrix are chosen.
- def `generate`
Generates a the graph.
- def `populate`
Populate graph with edges generated after a call to `DirectedPowerLawRandomGraph::generate`.
- def `writeEdges`
Write edges to file.

Public Attributes

- `graphSize`
Number of vertices to be considered for generation.
- `noOfEdges`
Number of edges to generate.
- `probA`
Parameters of the RMat algorithm.
- `probB`

Probability of choosing quadrant B.

- [probC](#)
Probability of choosing quadrant C.
- [probD](#)
Probability of choosing quadrant D.
- [serialEdgeList](#)
Temporary storage of edges.
- [debug](#)
Debug flag.
- [startVertX](#)
- [endVertX](#)
- [startVertY](#)
- [endVertY](#)

10.5.2 Member Function Documentation

10.5.2.1 def

**pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph::_-
_init__** (*self*, *size*, *noOfEdges*)

Constructs an empty graph.

Parameters:

size Number of vertices to be considered for generation
noOfEdges Number of edges to generate

10.5.2.2 def

pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph::setProbs
(*self*, *probA*, *probB*, *probC*, *probD*)

Sets the probability with which quadrants in an adjacency matrix are chosen.

Parameters:

probA Probability of choosing quadrant A
probB Probability of choosing quadrant B
probC Probability of choosing quadrant C
probD Probability of choosing quadrant D

Exceptions:

PackageExceptions::DistError

10.5.2.3 def
pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph::generate
(self, noOfThreads, noSelfLoops)

Generates a the graph.

Heart of web graph generation algorithm. Each thread gets an equal number of nodes to generate.

Parameters:

noOfThreads Number of threads to spawn for the graph generation. More threads does not correspond to fast generation

noSelfLoops If true (set to 1) self loops are discarded in the resulting graph.

10.5.2.4 def
pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph::populate
(self)

Populate graph with edges generated after a call to [DirectedPowerLawRandomGraph::generate](#).

You should call this method before you can use any of the non-overridden method in `Graph::NumberedEdgeDirectedGraph`

10.5.2.5 def
pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph::writeEdges
(self, fileName, format)

Write edges to file.

Parameters:

fileName File name to store edges

format Format of output file. Can take values:

'simple' = simple format

'dot' = format compatible with 'dot' command 'csv' = comma separated value format

Reimplemented from [pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph](#).

10.5.3 Member Data Documentation

10.5.3.1 pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph::probA

Parameters of the RMAT algorithm.

Decide the probability with which quadrants in an adjacency matrix are chosen

Todo

Add description about choosing these probabilities

Probability of choosing quadrant A

10.5

pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph

Class Reference

37

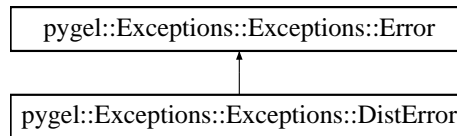
10.5.3.2 pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph::serialEdgeL

Temporary storage of edges.

Maintained for achieving performance

10.6 pygel::Exceptions::Exceptions::DistError Class Reference

Inheritance diagram for pygel::Exceptions::Exceptions::DistError::



10.6.1 Detailed Description

Represents a [DistError](#) exception.

It handles different types of probability distribution related exceptions

Public Member Functions

- `def __init__`
Constructs a [DistError](#) exception.

Public Attributes

- `message`
Error message

10.6.2 Member Function Documentation

10.6.2.1 `def pygel::Exceptions::Exceptions::DistError::__init__ (self, message)`

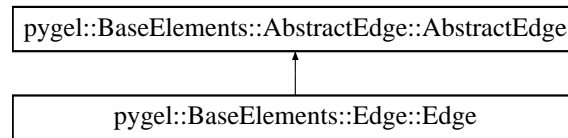
Constructs a [DistError](#) exception.

Parameters:

message Error message

10.7 pygel::BaseElements::Edge::Edge Class Reference

Inheritance diagram for pygel::BaseElements::Edge::Edge::



10.7.1 Detailed Description

Represents graph edge.

Public Member Functions

- def [__init__](#)
Constructs an empty edge.
- def [__init__](#)
Constructs a graph edge with given start and end vertices.
- def [getStartVertex](#)
Get the start vertex.
- def [getEndVertex](#)
Get the end vertex.
- def [setStartVertex](#)
Set the start vertex.
- def [setEndVertex](#)
Set the end vertex.

Public Attributes

- [startVertex](#)
Starting vertex of a edge of type BaseElements::Vertex.
- [endVertex](#)
Ending vertex of a edge of type BaseElements::Vertex.

10.7.2 Member Function Documentation

10.7.2.1 `def pygel::BaseElements::Edge::Edge::__init__ (self, startVertex, endVertex)`

Constructs a graph edge with given start and end vertices.

Parameters:

startVertex start vertex of the edge

endVertex end vertex of the edge

10.7.2.2 `def pygel::BaseElements::Edge::Edge::getStartVertex (self)`

Get the start vertex.

Returns:

startVertex Start vertex of type `BaseElements::Vertex`

10.7.2.3 `def pygel::BaseElements::Edge::Edge::getEndVertex (self)`

Get the end vertex.

Returns:

endVertex End vertex of type `BaseElements::Vertex`

10.7.2.4 `def pygel::BaseElements::Edge::Edge::setStartVertex (self, vertex)`

Set the start vertex.

Parameters:

startVertex Start vertex of type `BaseElements::Vertex`

10.7.2.5 `def pygel::BaseElements::Edge::Edge::setEndVertex (self, vertex)`

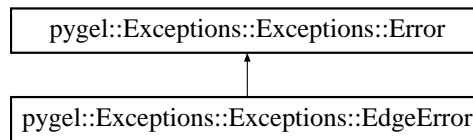
Set the end vertex.

Parameters:

endVertex End vertex of type `BaseElements::Vertex`

10.8 pygel::Exceptions::Exceptions::EdgeError Class Reference

Inheritance diagram for pygel::Exceptions::Exceptions::EdgeError::



10.8.1 Detailed Description

Represents a [EdgeError](#) exception.

It handles different types of graph edge related exceptions

Public Member Functions

- `def __init__`
Constructs a [EdgeError](#) exception.
- `def __init__`
Constructs a [EdgeError](#) exception.
- `def __init__`
Constructs a [EdgeError](#) exception.

Public Attributes

- `edgeNumber`
Vertex number for which the exception occurred.
- `message`
Error message
- `startVertexNumber`
Start vertex number of the edge.
- `endVertexNumber`
End vertex number of the edge.

10.8.2 Member Function Documentation

10.8.2.1 `def pygel::Exceptions::Exceptions::EdgeError::__init__(self, edgeNumber, message)`

Constructs a [EdgeError](#) exception.

Parameters:

edgeNumber Edge number for which the exception occurred
message Error message

10.8.2.2 `def pygel::Exceptions::Exceptions::EdgeError::__init__ (self, message)`

Constructs a [EdgeError](#) exception.

Parameters:

message Error message

10.8.2.3 `def pygel::Exceptions::Exceptions::EdgeError::__init__ (self, startVertexNumber, endVertexNumber, message)`

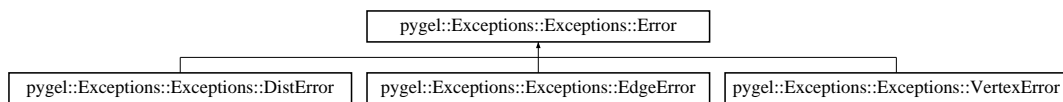
Constructs a [EdgeError](#) exception.

Parameters:

startVertexNumber startVertexNumber of the edge
endVertexNumber endVertexNumber of the edge
message Error message

10.9 pygel::Exceptions::Exceptions::Error Class Reference

Inheritance diagram for pygel::Exceptions::Exceptions::Error::



10.9.1 Detailed Description

Empty base class from which all exceptions are derived.

10.10 pygel::Exceptions::Exceptions::ErrorMessages Class Reference

10.10.1 Detailed Description

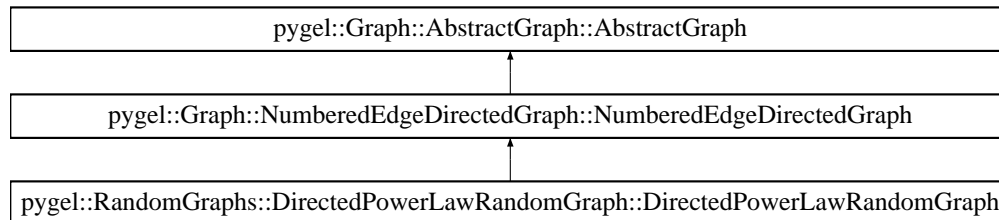
Collection of various error message strings.

Static Public Attributes

- string **vertexAlreadyExists** = 'Vertex number already exists'
- string **vertexNotFound** = 'Vertex number not found'
- string **edgeAlreadyExists** = 'Edge already exists'
- string **edgeNotFound** = 'Edge number not found'
- string **distAddOne** = 'Probabilities do not add to one'
- string **noSelfLoops** = 'No self loops are allowed for this graph'

10.11 pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph Class Reference

Inheritance diagram for pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::



10.11.1 Detailed Description

Represents a numbered edge graph.

Numbered edges are required to distinguish multiple edges between same set of vertices. This class also provides an indexed vertex and edge sets. These indices have certain advantages while computing in-degree and out-degree distributions.

Public Member Functions

- def [__init__](#)
Constructs a numbered edge graph.
- def [addEdge](#)
Adds an edge to a graph.
- def [deleteEdge](#)
Delete an edge.
- def [addVertex](#)
Adds a vertex.
- def [deleteVertex](#)
Deletes a vertex.
- def [getEdges](#)
Get all graph edges.
- def [getVertices](#)
Get all graph vertices.
- def [getLastEdgeNumber](#)
Get the last edge number.
- def [getOutNeighbors](#)
Get out-neighbors for a vertex.

- def [getInNeighbors](#)
Get in-neighbors for a vertex.
- def [getNumberOfOutNeighbors](#)
Get number of out-neighbors for a vertex.
- def [getNumberOfInNeighbors](#)
Get number of in-neighbors for a vertex.
- def [getNumberOfNeighbors](#)
Get number of neighbors for a vertex.
- def [getInDegreeDistribution](#)
Get in-degree distribution.
- def [getOutDegreeDistribution](#)
Get out-degree distribution.
- def [getJointDistribution](#)
Get joint-degree distribution.
- def [getDegreeDistribution](#)
Get degree distribution.
- def [getVerticesByInDegree](#)
Gets all the vertices with a particular in-degree.
- def [getVerticesByOutDegree](#)
Gets all the vertices with a particular out-degree.
- def [getSCComponents](#)
Gets the strongly connected components of a graph.
- def [getOutComponent](#)
Gives the out component for a strongly connected component.
- def [writeCC](#)
Write the connected components to a file.
- def [writeEdges](#)
Write edges to file.
- def [readEdges](#)
Read edges from file.
- def [findEdge](#)
Find edge with a given edge number.
- def [findVertex](#)

Find vertex with a given vertex number.

- def [hasVertex](#)

Checks if vertex is present.

Public Attributes

- [edgeIndex](#)

Dictionary of edges, indexed by edge number.

- [vertexIndex](#)

Dictionary of vertices, indexed by vertex number.

- [parentIndex](#)

Dictionary of vertices, indexed by parent.

- [parentEdgeIndex](#)

Dictionary of vertices and edge numbers, indexed by parent.

10.11.2 Member Function Documentation

10.11.2.1 def pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::addEdge(*self*, *edge*)

Adds an edge to a graph.

It also updates the vertex and edge indices.

Parameters:

edge Edge of type BaseElements::Edge to be added to the graph

10.11.2.2 def pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::deleteEdge(*self*, *edgeNumber*)

Delete an edge.

Parameters:

edgeNumber Edge number to be deleted

10.11.2.3 def
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::addVertex
(*self*, *vertexNumber*)

Adds a vertex.

Should be used with care

Parameters:

vertexNumber Vertex number of vertex to be added

Exceptions:

PackageExceptions::VertexError

10.11.2.4 def
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::deleteVertex
(*self*, *vertexNumber*)

Deletes a vertex.

Should be used with care

Parameters:

vertexNumber Vertex number to be deleted

10.11.2.5 def
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getEdges
(*self*)

Get all graph edges.

Returns:

edgeIndex Dictionary of edges, indexed by edge number

10.11.2.6 def
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getVertices
(*self*)

Get all graph vertices.

Returns:

vertexIndex Dictionary of vertices, indexed by vertex number

10.11.2.7 def
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getLastEdgeNumber
(*self*)

Get the last edge number.

Returns:

`__lastEdgeNumber` the last edge number assigned to edges.

10.11.2.8 def
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getOutNeighbors
(*self*, *vertexNumber*)

Get out-neighbors for a vertex.

Parameters:

vertexNumber Vertex number for which out-neighbors have to be obtained

Returns:

`outNeighbors` List of out-neighbors. Each element of type `BaseElements::Vertex`

10.11.2.9 def
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getInNeighbors
(*self*, *vertexNumber*)

Get in-neighbors for a vertex.

Parameters:

vertexNumber Vertex number for which in-neighbors have to be obtained

Returns:

`inNeighbors` List of in-neighbors. Each element of type `BaseElements::Vertex`

10.11.2.10 def
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getNumberOfOutNeighbors
(*self*, *vertexNumber*)

Get number of out-neighbors for a vertex.

Parameters:

vertexNumber Vertex number for which number of out-neighbors have to be obtained

Returns:

Number of out-neighbors

10.11.2.11 **def**
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getNumberOfInNeighbors
(*self*, *vertexNumber*)

Get number of in-neighbors for a vertex.

Parameters:

vertexNumber Vertex number for which number of in-neighbors have to be obtained

Returns:

Number of in-neighbors

10.11.2.12 **def**
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getNumberOfNeighbors
(*self*, *vertexNumber*)

Get number of neighbors for a vertex.

Parameters:

vertexNumber Vertex number for which number of neighbors have to be obtained

Returns:

Number of neighbors

10.11.2.13 **def**
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getInDegreeDistribution
(*self*)

Get in-degree distribution.

Returns:

inDegreeDistribution Dictionary indexed on in-degree. Values are the number of nodes for a in-degree

10.11.2.14 **def**
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getOutDegreeDistribution
(*self*)

Get out-degree distribution.

Returns:

outDegreeDistribution Dictionary indexed on in-degree. Values are the number of nodes for a out-degree

10.11.2.15 def
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getJointDistribution
(*self*)

Get joint-degree distribution.

Returns:

jointDegreeDistribution Dictionary indexed on out-degree and in-degree. Values are the number of nodes for a given combination of out-degree and in-degree

10.11.2.16 def
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getDegreeDistribution
(*self*)

Get degree distribution.

Returns:

degreeDistribution Dictionary indexed on degree. Values are the number of nodes for a degree

10.11.2.17 def
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getVerticesByInDegree
(*self*, *degree*)

Gets all the vertices with a particular in-degree.

Parameters:

degree In-degree to look for

Returns:

degreeNodes List of vertices. Each element of type BaseElements::Vertex

10.11.2.18 def
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getVerticesByOutDegree
(*self*, *degree*)

Gets all the vertices with a particular out-degree.

Parameters:

degree Out-degree to look for

Returns:

degreeNodes List of vertices. Each element of type BaseElements::Vertex

10.11.2.19 **def**
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getSCComponents
(self, getLargest)

Gets the strongly connected components of a graph.

It uses [Tarjan's strongly connected components algorithm](#).

Parameters:

getLargest If greater than 0, only returns the largest connected component

Returns:

allSCC List of a List of connected components

10.11.2.20 **def**
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::getOutComponent
(self, stronglyCC)

Gives the out component for a strongly connected component.

Parameters:

stronglyCC Strongly connected component for which th out-component is to be determined

Returns:

outComponent List of vertices in the out component

10.11.2.21 **def**
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::writeCC
(self, fileName, allSCC)

Write the connected components to a file.

Parameters:

fileName File name to store the connected components

allSCC List of list of connected components

10.11.2.22 **def**
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::writeEdges
(self, fileName, format)

Write edges to file.

Parameters:

fileName File name to store edges in

format Format of output file. Can take values:

'simple' = simple format

'dot' = format compatible with 'dot' command

Reimplemented in [pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph](#).

10.11.2.23 **def**
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::readEdges
(*self*, *fileName*, *format*)

Read edges from file.

Parameters:

fileName File name to read edges from
format Format of input file. Can take values:
 'simple' = simple format

10.11.2.24 **def**
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::findEdge
(*self*, *edgeNumber*)

Find edge with a given edge number.

Parameters:

edgeNumber Edge number to look for

Exceptions:

PackageExceptions::EdgeError

Returns:

Matched edge of type BaseElements::Edge

10.11.2.25 **def**
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::findVertex
(*self*, *vertexNumber*)

Find vertex with a given vertex number.

Parameters:

vertexNumber Vertex number to look for

Exceptions:

PackageExceptions::VertexError

Returns:

Matched vertex of type BaseElements::Vertex

10.11.2.26 **def**
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph::hasVertex
(*self*, *vertexNumber*)

Checks if vertex is present.

Parameters:

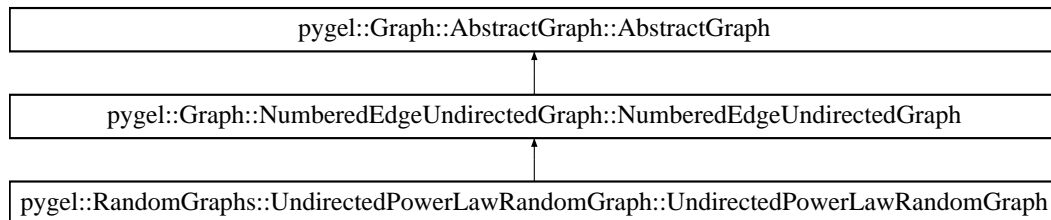
vertexNumber Vertex number of the vertex to check

Returns:

0 if found. 1 if not found

10.12 pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph Class Reference

Inheritance diagram for pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::



10.12.1 Detailed Description

Represents a numbered edge graph.

Numbered edges are required to distinguish multiple edges between same set of vertices. This class also provides an indexed vertex and edge sets. These indices have certain advantages while computing in-degree and out-degree distributions.

Public Member Functions

- def [__init__](#)
Constructs a numbered edge graph.
- def [addEdge](#)
Adds an edge to a graph.
- def [deleteEdge](#)
Delete an edge.
- def [addVertex](#)
Adds a vertex.
- def [deleteVertex](#)
Deletes a vertex.
- def [getEdges](#)
Get all graph edges.
- def [getVertices](#)
Get all graph vertices.
- def [getVertexNumbers](#)
Get all graph vertex numbers.
- def [getLastEdgeNumber](#)
Get the last edge number.

- def [getNeighbors](#)
Get neighbors for a vertex.
- def [getNumberOfNeighbors](#)
Get number of neighbors for a vertex.
- def [getDegreeDistribution](#)
Get degree distribution.
- def [getSCComponents](#)
Gets the strongly connected components of a graph.
- def [writeCC](#)
Write the connected components to a file.
- def [writeEdges](#)
Write edges to file.
- def [readEdges](#)
Read edges from file.
- def [findEdge](#)
Find edge with a given edge number.
- def [findVertex](#)
Find vertex with a given vertex number.
- def [hasVertex](#)
Checks if vertex is present.

Public Attributes

- [edgeIndex](#)
Dictionary of edges, indexed by edge number.
- [vertexIndex](#)
Dictionary of vertices, indexed by vertex number.
- [parentIndex](#)
Dictionary of vertices, indexed by parent.
- [parentEdgeIndex](#)
Dictionary of vertices and edge numbers, indexed by parent.
- [logger](#)
Logger instance.

10.12.2 Member Function Documentation

10.12.2.1 def

```
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::addEdge  
( self, edge)
```

Adds an edge to a graph.

It also updates the vertex and edge indices.

Parameters:

edge Edge of type BaseElements::Edge to be added to the graph

10.12.2.2 def

```
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::deleteEdge  
( self, edgeNumber)
```

Delete an edge.

Parameters:

edgeNumber Edge number to be deleted

10.12.2.3 def

```
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::addVertex  
( self, vertexNumber)
```

Adds a vertex.

Should be used with care

Parameters:

vertexNumber Vertex number of vertex to be added

Exceptions:

PackageExceptions::VertexError

10.12.2.4 def

```
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::deleteVertex  
( self, vertexNumber)
```

Deletes a vertex.

Should be used with care

Parameters:

vertexNumber Vertex number to be deleted

10.12.2.5 **def**
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::getEdges
(*self*)

Get all graph edges.

Returns:

edgeIndex Dictionary of edges, indexed by edge number

10.12.2.6 **def**
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::getVertices
(*self*)

Get all graph vertices.

Returns:

vertexIndex Dictionary of vertices, indexed by vertex number

10.12.2.7 **def**
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::getVertexNumbers
(*self*)

Get all graph vertex numbers.

Returns:

an array consisting of all the vertices numbers

10.12.2.8 **def**
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::getLastEdgeNumber
(*self*)

Get the last edge number.

Returns:

__lastEdgeNumber the last edge number assigned to edges.

10.12.2.9 **def**
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::getNeighbors
(*self*, *vertexNumber*)

Get neighbors for a vertex.

Parameters:

vertexNumber Vertex number for which neighbors have to be obtained

Returns:

neighbors List of neighbors. Each element of type BaseElements::Vertex

10.12.2.10 def
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::getNumberOfNeighbors
(*self*, *vertexNumber*)

Get number of neighbors for a vertex.

Parameters:

vertexNumber Vertex number for which number of neighbors have to be obtained

Returns:

Number of neighbors

10.12.2.11 def
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::getDegreeDistribution
(*self*)

Get degree distribution.

Returns:

degreeDistribution Dictionary indexed on degree. Values are the number of nodes for a degree

10.12.2.12 def
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::getSCComponents
(*self*, *getLargest*)

Gets the strongly connected components of a graph.

It uses [Tarjan's strongly connected components algorithm](#).

Parameters:

getLargest If greater than 0, only returns the largest connected component

Returns:

allSCC List of a List of connected components

10.12.2.13 def
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::writeCC
(*self*, *fileName*, *allSCC*)

Write the connected components to a file.

Parameters:

fileName File name to store the connected components

allSCC List of list of connected components

10.12.2.14 **def**
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::writeEdges
(*self*, *fileName*, *format*)

Write edges to file.

Parameters:

fileName File name to store edges in

format Format of output file. Can take values:

'simple' = simple format

'dot' = format compatible with 'dot' command

Reimplemented in [pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph](#).

10.12.2.15 **def**
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::readEdges
(*self*, *fileName*, *format*)

Read edges from file.

Parameters:

fileName File name to read edges from

format Format of input file. Can take values:

'simple' = simple format

10.12.2.16 **def**
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::findEdge
(*self*, *edgeNumber*)

Find edge with a given edge number.

Parameters:

edgeNumber Edge number to look for

Exceptions:

PackageExceptions::EdgeError

Returns:

Matched edge of type BaseElements::Edge

10.12.2.17 **def**
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::findVertex
(*self*, *vertexNumber*)

Find vertex with a given vertex number.

Parameters:

vertexNumber Vertex number to look for

Exceptions:

PackageExceptions::VertexError

Returns:

Matched vertex of type BaseElements::Vertex

10.12.2.18 **def**
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph::hasVertex
(*self*, *vertexNumber*)

Checks if vertex is present.

Parameters:

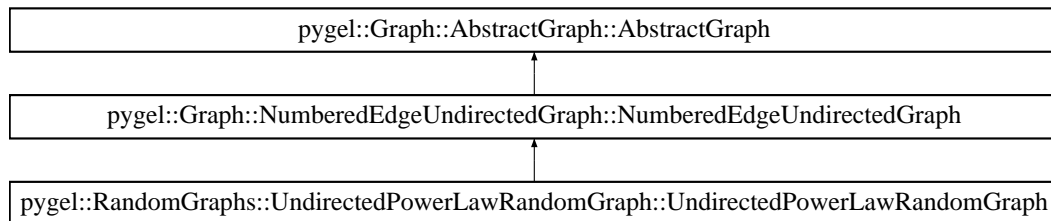
vertexNumber Vertex number of the vertex to check

Returns:

0 if found. 1 if not found

10.13 pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph Class Reference

Inheritance diagram for pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph::



10.13.1 Detailed Description

Generates a synthetic Web graph or Power Law graph using an RMat algorithm.

Public Member Functions

- def `__init__`
Constructs an empty graph.
- def `setProbs`
Sets the probability with which quadrants in an adjacency matrix are chosen.
- def `generate`
Generates a the graph.
- def `populate`
Populate graph with edges generated after a call to `DirectedPowerLawRandomGraph::generate`.
- def `writeEdges`
Write edges to file.

Public Attributes

- `graphSize`
Number of vertices to be considered for generation.
- `noOfEdges`
Number of edges to generate.
- `probA`
Parameters of the RMat algorithm.
- `probB`

Probability of choosing quadrant B.

- [probC](#)
Probability of choosing quadrant C.
- [probD](#)
Probability of choosing quadrant D.
- [serialEdgeList](#)
Temporary storage of edges.
- [debug](#)
Debug flag.
- [startVertX](#)
- [endVertX](#)
- [startVertY](#)
- [endVertY](#)
- [logger](#)
Logger instance.

10.13.2 Member Function Documentation

10.13.2.1 `def`

`pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph::_init__ (self, size, noOfEdges)`

Constructs an empty graph.

Parameters:

size Number of vertices to be considered for generation
noOfEdges Number of edges to generate

10.13.2.2 `def`

`pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph::setProb (self, probA, probB, probC, probD)`

Sets the probability with which quadrants in an adjacency matrix are chosen.

Parameters:

probA Probability of choosing quadrant A
probB Probability of choosing quadrant B
probC Probability of choosing quadrant C
probD Probability of choosing quadrant D

Exceptions:

PackageExceptions::DistError

10.13.2.3 **def**
pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph::generate
(self, noOfThreads, noSelfLoops)

Generates a the graph.

Heart of web graph generation algorithm. Each thread gets an equal number of nodes to generate.

Parameters:

noOfThreads Number of threads to spawn for the graph generation. More threads does not correspond to fast generation

noSelfLoops If true (set to 1) self loops are discarded in the resulting graph.

10.13.2.4 **def**
pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph::populate
(self)

Populate graph with edges generated after a call to DirectedPowerLawRandomGraph::generate.

You should call this method before you can use any of the non-overridden method in Graph::NumberedEdgeDirectedGraph

10.13.2.5 **def**
pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph::writeEdges
(self, fileName, format)

Write edges to file.

Parameters:

fileName File name to store edges

format Format of output file. Can take values:

'simple' = simple format

'dot' = format compatible with 'dot' command 'csv' = comma separated value format

Reimplemented from [pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph](#).

10.13.3 Member Data Documentation

10.13.3.1 **pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph::probA**

Parameters of the RMAT algorithm.

Decide the probability with which quadrants in an adjacency matrix are chosen

Todo

Add description about choosing these probabilities

Probability of choosing quadrant A

10.13

pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph
Class Reference 65

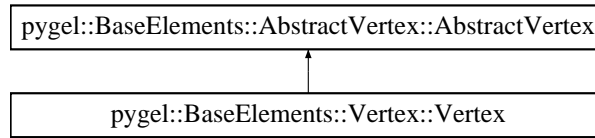
10.13.3.2 **pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph::serial**

Temporary storage of edges.

Maintained for achieving performance

10.14 pygel::BaseElements::Vertex::Vertex Class Reference

Inheritance diagram for pygel::BaseElements::Vertex::Vertex::



10.14.1 Detailed Description

Represents graph vertex.

Public Member Functions

- def [__init__](#)
Constructs graph vertex given a vertex number.
- def [getVertexNumber](#)
Get vertex number.
- def [setVertexNumber](#)
Set vertex number.

Public Attributes

- [vertexNumber](#)
Vertex number.

10.14.2 Member Function Documentation

10.14.2.1 def pygel::BaseElements::Vertex::Vertex::__init__ (self, vertexNumber)

Constructs graph vertex given a vertex number.

Parameters:

vertexNumber [Vertex](#) number to be assigned to the created vertex

10.14.2.2 def pygel::BaseElements::Vertex::Vertex::getVertexNumber (self)

Get vertex number.

Returns:

vertexNumber [Vertex](#) number of this vertex

10.14.2.3 def pygel::BaseElements::Vertex::Vertex::setVertexNumber (*self*, *vertexNumber*)

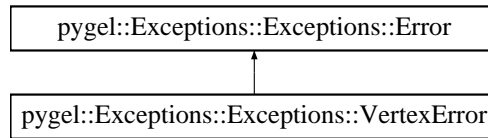
Set vertex number.

Parameters:

vertexNumber New vertex number

10.15 pygel::Exceptions::Exceptions::VertexError Class Reference

Inheritance diagram for pygel::Exceptions::Exceptions::VertexError::



10.15.1 Detailed Description

Represents a [VertexError](#) exception.

It handles different types of graph vertex related exceptions

Public Member Functions

- def [__init__](#)
Constructs a [VertexError](#) exception.

Public Attributes

- [vertexNumber](#)
Vertex number for which the exception occurred.
- [message](#)
Error message

10.15.2 Member Function Documentation

10.15.2.1 def pygel::Exceptions::Exceptions::VertexError::__init__ (*self*, *vertexNumber*, *message*)

Constructs a [VertexError](#) exception.

Parameters:

- vertexNumber* Vertex number for which the exception occurred
message Error message

10.16 pygel::BaseElements::WeightedVertex::WeightedVertex Class Reference

10.16.1 Detailed Description

Represents a weighted vertex.

Public Member Functions

- def `__init__`
Constructs a weighted vertex given a vertex number and vertex weight.
- def `getWeight`
Get vertex weight.
- def `setWeight`
Set vertex weight.

Public Attributes

- `vertexWeight`
Vertex weight.

10.16.2 Member Function Documentation

10.16.2.1 def pygel::BaseElements::WeightedVertex::WeightedVertex::__init__ (*self*, *vertexNumber*, *vertexWeight*)

Constructs a weighted vertex given a vertex number and vertex weight.

Parameters:

vertexNumber vertex number to be assigned to the created vertex

vertexWeight vertex weight to be assigned to the created vertex

10.16.2.2 def pygel::BaseElements::WeightedVertex::WeightedVertex::getWeight (*self*)

Get vertex weight.

Returns:

vertexWeight Vertex weight of this vertex

10.16.2.3 `def pygel::BaseElements::WeightedVertex::WeightedVertex::setWeight (self,
vertexWeight)`

Set vertex weight.

Parameters:

vertexWeight New vertex weight

10.17 pygel::BaseElements::WeightedVertices::WeightedVertices Class Reference

10.17.1 Detailed Description

Represents a collection of weighted vertices of type BaseElements::WeightedVertices.

Public Member Functions

- def [__init__](#)
Initialize an empty collection.
- def [addVertex](#)
Add vertex to the collection.
- def [delVertex](#)
Delete vertex from the collection.
- def [getVertices](#)
Get all vertices from the collection.
- def [findVertex](#)
Find vertex in the collection.
- def [findWeight](#)
Find weight of a given vertex.
- def [hasVertex](#)
Checks if vertex is present.

Public Attributes

- [weightedVertices](#)
A dictionary of weighted vertices indexed by vertex numbers and values of type BaseElements::WeightedVertex.

10.17.2 Member Function Documentation

10.17.2.1 def pygel::BaseElements::WeightedVertices::WeightedVertices::__init__ (self)

Initialize an empty collection.

10.17.2.2 def pygel::BaseElements::WeightedVertices::WeightedVertices::addVertex (self, weightedVertex)

Add vertex to the collection.

Parameters:

weightedVertex Weighted vertex to be added. Should be of type BaseElements::WeightedVertex

**10.17.2.3 def pygel::BaseElements::WeightedVertices::WeightedVertices::delVertex (*self*,
vertexNumber)**

Delete vertex from the collection.

Parameters:

vertexNumber Vertex number of the vertex to be deleted

10.17.2.4 def pygel::BaseElements::WeightedVertices::WeightedVertices::getVertices (*self*)

Get all vertices from the collection.

Returns:

weightedVertices A dict of all weighted vertices indexed by vertex number

**10.17.2.5 def pygel::BaseElements::WeightedVertices::WeightedVertices::findVertex (*self*,
vertexNumber)**

Find vertex in the collection.

Parameters:

vertexNumber Vertex number to be found

Exceptions:

PackageExceptions::VertexError

Returns:

weightedVertex Found weighted vertex of type BaseElements::WeightedVertex

**10.17.2.6 def pygel::BaseElements::WeightedVertices::WeightedVertices::findWeight (*self*,
vertexNumber)**

Find weight of a given vertex.

Parameters:

vertexNumber Vertex number of the vertex whose weight is to be found

Returns:

vertexWeight Weight of vertex

10.17.2.7 `def pygel::BaseElements::WeightedVertices::WeightedVertices::hasVertex (self,
vertexNumber)`

Checks if vertex is present.

Parameters:

vertexNumber Vertex number to be checked

Returns:

0 if vertex is found. Otherwise 1

Index

`__init__`
 pygel::BaseElements::Edge::Edge, 40
 pygel::BaseElements::Vertex::Vertex, 66
 pygel::BaseElements::WeightedVertex::WeightedVertex, 69
 pygel::BaseElements::WeightedVertices::WeightedVertices, 71
 pygel::Exceptions::Exceptions::DistError, 38
 pygel::Exceptions::Exceptions::EdgeError, 41, 42
 pygel::Exceptions::Exceptions::VertexError, 68
 pygel::RandomGraphs::ChooseEdges::ChooseEdges, 32
 pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph, 35
 pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph, 63
addEdge
 pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 47
 pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 57
addVertex
 pygel::BaseElements::WeightedVertices::WeightedVertices, 71
 pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 47
 pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 57
BaseElements, 21
Basic Elements, 15
deleteEdge
 pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 47
 pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 57
deleteVertex
 pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 48
 pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 57
delVertex
 pygel::BaseElements::WeightedVertices::WeightedVertices, 72
 endVertX
 pygel::RandomGraphs::ChooseEdges::ChooseEdges, 32
 endVertY
 pygel::RandomGraphs::ChooseEdges::ChooseEdges, 33
 Exceptions, 22
 findEdge
 pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 53
 pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 60
 endVertex
 pygel::BaseElements::WeightedVertices::WeightedVertices, 72
 pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 53
 pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 60
 findWeight
 pygel::BaseElements::WeightedVertices::WeightedVertices, 72
 generate
 pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph, 35
 pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph, 63
 getDegreeDistribution
 pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 51
 pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 59
 getEdges
 pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 48
 pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 57
 getEndVertex
 pygel::BaseElements::AbstractEdge::AbstractEdge, 28

pygel::BaseElements::Edge::Edge, 40	pygel::BaseElements::WeightedVertices::WeightedVertices, 72
getInDegreeDistribution	pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 48
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 50	pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 58
getInNeighbors	pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 49
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 49	getVerticesByInDegree
getJointDistribution	pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 50
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 50	getVerticesByOutDegree
getLastEdgeNumber	pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 48
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 48	getWeight
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 58	pygel::BaseElements::WeightedVertex::WeightedVertex, 69
getNeighbors	Graph, 17, 23
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 58	hasVertex
getNumberOfInNeighbors	pygel::BaseElements::WeightedVertices::WeightedVertices, 72
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 49	pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 53
getNumberOfNeighbors	pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 50
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 50	NumberOfEdges
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 58	pygel::RandomGraphs::ChooseEdges::ChooseEdges, 33
getNumberOfOutNeighbors	NumberOfLoops
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 49	pygel::RandomGraphs::ChooseEdges::ChooseEdges, 33
getOutComponent	pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 52
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 52	Package Exceptions, 16
getOutDegreeDistribution	PackageExceptions, 24
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 50	populate
getOutNeighbors	pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph, 36
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 49	pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph, 64
getSCComponents	probA
pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 51	pygel::RandomGraphs::ChooseEdges::ChooseEdges, 33
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 59	pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph, 36
getStartVertex	pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph, 64
pygel::BaseElements::AbstractEdge::AbstractEdge, 28	probB
pygel::BaseElements::Edge::Edge, 40	pygel::RandomGraphs::ChooseEdges::ChooseEdges, 33
getVertexNumber	probC
pygel::BaseElements::Vertex::Vertex, 66	pygel::RandomGraphs::ChooseEdges::ChooseEdges, 33
getVertexNumbers	33
pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 58	pygel::RandomGraphs::ChooseEdges::ChooseEdges, 33
getVertices	33

- pygel::BaseElements::AbstractEdge::AbstractEdge, 27
 - getEndVertex, 28
 - getStartVertex, 28
 - setEndVertex, 28
 - setStartVertex, 28
- pygel::BaseElements::AbstractVertex::AbstractVertex, 30
- pygel::BaseElements::Edge::Edge, 39
 - __init__, 40
 - getEndVertex, 40
 - getStartVertex, 40
 - setEndVertex, 40
 - setStartVertex, 40
- pygel::BaseElements::Vertex::Vertex, 66
 - __init__, 66
 - getVertexNumber, 66
 - setVertexNumber, 66
- pygel::BaseElements::WeightedVertex::WeightedVertex, 69
 - __init__, 69
 - getWeight, 69
 - setWeight, 69
- pygel::BaseElements::WeightedVertices::WeightedVertices, 71
 - __init__, 71
 - addVertex, 71
 - delVertex, 72
 - findVertex, 72
 - findWeight, 72
 - getVertices, 72
 - hasVertex, 72
- pygel::Exceptions::Exceptions::DistError, 38
 - __init__, 38
- pygel::Exceptions::Exceptions::EdgeError, 41
 - __init__, 41, 42
- pygel::Exceptions::Exceptions::Error, 43
- pygel::Exceptions::Exceptions::ErrorMessages, 44
- pygel::Exceptions::Exceptions::VertexError, 68
 - __init__, 68
- pygel::Graph::AbstractGraph::AbstractGraph, 29
- pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 45
 - addEdge, 47
 - addVertex, 47
 - deleteEdge, 47
 - deleteVertex, 48
 - findEdge, 53
 - findVertex, 53
 - getDegreeDistribution, 51
 - getEdges, 48
 - getInDegreeDistribution, 50
 - getInNeighbors, 49
 - getJointDistribution, 50
 - getLastEdgeNumber, 48
 - getNumberOfInNeighbors, 49
 - getNumberOfNeighbors, 50
 - getNumberOfOutNeighbors, 49
 - getOutComponent, 52
 - getOutDegreeDistribution, 50
 - getOutNeighbors, 49
 - getSCComponents, 51
 - getVertices, 48
 - getVerticesByInDegree, 51
 - getVerticesByOutDegree, 51
 - hasVertex, 53
 - readEdges, 52
 - writeCC, 52
 - writeEdges, 52
- pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 55
 - addEdge, 57
 - addVertex, 57
 - deleteEdge, 57
 - deleteVertex, 57
 - findEdge, 60
 - findVertex, 60
 - getDegreeDistribution, 59
 - getEdges, 57
 - getLastEdgeNumber, 58
 - getNeighbors, 58
 - getNumberOfNeighbors, 58
 - getSCComponents, 59
 - getVertexNumbers, 58
 - getVertices, 58
 - hasVertex, 61
 - readEdges, 60
 - writeCC, 59
 - writeEdges, 59
- pygel::RandomGraphs::ChooseEdges::ChooseEdges, 31
 - __init__, 32
 - endVertX, 32
 - endVertY, 33
 - numberOfEdges, 33
 - probA, 33
 - probB, 33
 - probC, 33
 - probD, 33
 - selectVertex, 32
 - serialEdgeList, 32
 - startVertX, 32
 - startVertY, 32
- pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph, 34
 - __init__, 35
 - generate, 35

- populate, 36
- probA, 36
- serialEdgeList, 36
- setProbs, 35
- writeEdges, 36
- pygel::RandomGraphs::UndirectedPowerLawRandomGraph
 - 62
 - __init__, 63
 - generate, 63
 - populate, 64
 - probA, 64
 - serialEdgeList, 64
 - setProbs, 63
 - writeEdges, 64
- Random Graphs, 18
- RandomGraphs, 25
- readEdges
 - pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 52
 - pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 60
- selectVertex
 - pygel::RandomGraphs::ChooseEdges::ChooseEdges, 32
- serialEdgeList
 - pygel::RandomGraphs::ChooseEdges::ChooseEdges, 32
 - pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph, 36
 - pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph, 64
- setEndVertex
 - pygel::BaseElements::AbstractEdge::AbstractEdge, 28
 - pygel::BaseElements::Edge::Edge, 40
- setProbs
 - pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph, 35
 - pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph, 63
- setStartVertex
 - pygel::BaseElements::AbstractEdge::AbstractEdge, 28
 - pygel::BaseElements::Edge::Edge, 40
- setVertexNumber
 - pygel::BaseElements::Vertex::Vertex, 66
- setWeight
 - pygel::BaseElements::WeightedVertex::WeightedVertex, 69
- startVertX
 - pygel::RandomGraphs::ChooseEdges::ChooseEdges, 32
- startVertY
 - pygel::RandomGraphs::ChooseEdges::ChooseEdges, 32
- System, 19, 26
- writeCG
 - pygel::UndirectedPowerLawRandomGraph, 52
 - pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 59
 - pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 59
- writeEdges
 - pygel::Graph::NumberedEdgeDirectedGraph::NumberedEdgeDirectedGraph, 52
 - pygel::Graph::NumberedEdgeUndirectedGraph::NumberedEdgeUndirectedGraph, 59
 - pygel::RandomGraphs::DirectedPowerLawRandomGraph::DirectedPowerLawRandomGraph, 36
 - pygel::RandomGraphs::UndirectedPowerLawRandomGraph::UndirectedPowerLawRandomGraph, 64