

Python Webgraph Generator Reference Manual

Generated by Doxygen 1.4.6

Wed Aug 1 14:12:27 2007

Contents

1	Python Webgraph Generator Main Page	1
1.1	Description	1
1.2	Todo Lists	1
2	Python Webgraph Generator Module Index	3
2.1	Python Webgraph Generator Modules	3
3	Python Webgraph Generator Namespace Index	5
3.1	Python Webgraph Generator Namespace List	5
4	Python Webgraph Generator Hierarchical Index	7
4.1	Python Webgraph Generator Class Hierarchy	7
5	Python Webgraph Generator Class Index	9
5.1	Python Webgraph Generator Class List	9
6	Python Webgraph Generator Page Index	11
6.1	Python Webgraph Generator Related Pages	11
7	Python Webgraph Generator Module Documentation	13
7.1	Basic Elements	13
7.2	Package Exceptions	14
7.3	Graph	15
7.4	Random Graphs	16
8	Python Webgraph Generator Namespace Documentation	17
8.1	BaseElements Namespace Reference	17
8.2	Exceptions Namespace Reference	18
8.3	Graph Namespace Reference	19
8.4	PackageExceptions Namespace Reference	20
8.5	RandomGraphs Namespace Reference	21

9 Python Webgraph Generator Class Documentation	23
9.1 pygel::BaseElements::AbstractEdge::AbstractEdge Class Reference	23
9.2 pygel::Graph::AbstractGraph::AbstractGraph Class Reference	24
9.3 pygel::BaseElements::AbstractVertex::AbstractVertex Class Reference	25
9.4 pygel::RandomGraphs::ChooseEdges::ChooseEdges Class Reference	26
9.5 pygel::Exceptions::Exceptions::DistError Class Reference	29
9.6 pygel::BaseElements::Edge::Edge Class Reference	30
9.7 pygel::Exceptions::Exceptions::EdgeError Class Reference	32
9.8 pygel::Exceptions::Exceptions::Error Class Reference	33
9.9 pygel::Exceptions::Exceptions::ErrorMessage Class Reference	34
9.10 pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph Class Reference	35
9.11 pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph Class Reference	41
9.12 pygel::BaseElements::Vertex::Vertex Class Reference	44
9.13 pygel::Exceptions::Exceptions::VertexError Class Reference	45
9.14 pygel::BaseElements::WeightedVertex::WeightedVertex Class Reference	46
9.15 pygel::BaseElements::WeightedVertices::WeightedVertices Class Reference	48
10 Python Webgraph Generator Page Documentation	51
10.1 General Todo List	51
10.2 Todo List	52

Chapter 1

Python Webgraph Generator 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

Python Webgraph Generator Module Index

2.1 Python Webgraph Generator Modules

Here is a list of all modules:

Basic Elements	13
Package Exceptions	14
Graph	15
Random Graphs	16

Chapter 3

Python Webgraph Generator Namespace Index

3.1 Python Webgraph Generator Namespace List

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

BaseElements (Module for defining basic elements)	17
Exceptions (Module for handling exceptions)	18
Graph (Module for handling graphs)	19
PackageExceptions (A module handling package exceptions)	20
RandomGraphs (Module for handling random graphs)	21

Chapter 4

Python Webgraph Generator Hierarchical Index

4.1 Python Webgraph Generator Class Hierarchy

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

pygel::BaseElements::AbstractEdge::AbstractEdge	23
pygel::Graph::AbstractGraph::AbstractGraph	24
pygel::BaseElements::AbstractVertex::AbstractVertex	25
pygel::RandomGraphs::ChooseEdges::ChooseEdges	26
pygel::BaseElements::Edge::Edge	30
pygel::Exceptions::Exceptions::Error	33
pygel::Exceptions::Exceptions::DistError	29
pygel::Exceptions::Exceptions::EdgeError	32
pygel::Exceptions::Exceptions::VertexError	45
pygel::Exceptions::Exceptions::ErrorMessage	34
pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph	35
pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph	41
pygel::BaseElements::Vertex::Vertex	44
pygel::BaseElements::WeightedVertex::WeightedVertex	46
pygel::BaseElements::WeightedVertices::WeightedVertices	48

Chapter 5

Python Webgraph Generator Class Index

5.1 Python Webgraph Generator Class List

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

<code>pygel::BaseElements::AbstractEdge::AbstractEdge</code> (Abstract class for representing an edge) . . .	23
<code>pygel::Graph::AbstractGraph::AbstractGraph</code> (Abstract class for representing a graph)	24
<code>pygel::BaseElements::AbstractVertex::AbstractVertex</code> (Abstract class for representing a vertex)	25
<code>pygel::RandomGraphs::ChooseEdges::ChooseEdges</code> (Thread for selecting a set of edges)	26
<code>pygel::Exceptions::Exceptions::DistError</code> (Represents a <code>DistError</code> exception)	29
<code>pygel::BaseElements::Edge::Edge</code> (Represents graph edge)	30
<code>pygel::Exceptions::Exceptions::EdgeError</code> (Represents a <code>EdgeError</code> exception)	32
<code>pygel::Exceptions::Exceptions::Error</code> (Empty base class from which all exceptions are derived)	33
<code>pygel::Exceptions::Exceptions::ErrorMessage</code> (Collection of various error message strings) . .	34
<code>pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph</code> (Represents a numbered edge graph)	35
<code>pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph</code> (Generates a synthetic Web graph or Power Law graph using an RMAT algorithm)	41
<code>pygel::BaseElements::Vertex::Vertex</code> (Represents graph vertex)	44
<code>pygel::Exceptions::Exceptions::VertexError</code> (Represents a <code>VertexError</code> exception)	45
<code>pygel::BaseElements::WeightedVertex::WeightedVertex</code> (Represents a weighted vertex)	46
<code>pygel::BaseElements::WeightedVertices::WeightedVertices</code> (Represents a collection of weighted vertices of type <code>BaseElements::WeightedVertices</code>)	48

Chapter 6

Python Webgraph Generator Page Index

6.1 Python Webgraph Generator Related Pages

Here is a list of all related documentation pages:

General Todo List	51
Todo List	52

Chapter 7

Python Webgraph Generator Module Documentation

7.1 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`.

7.2 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.

7.3 Graph

Classes

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

7.4 Random Graphs

Classes

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

Chapter 8

Python Webgraph Generator Namespace Documentation

8.1 BaseElements Namespace Reference

8.1.1 Detailed Description

Module for defining basic elements.

8.2 Exceptions Namespace Reference

8.2.1 Detailed Description

Module for handling exceptions.

8.3 Graph Namespace Reference

8.3.1 Detailed Description

Module for handling graphs.

8.4 PackageExceptions Namespace Reference

8.4.1 Detailed Description

A module handling package exceptions.

8.5 RandomGraphs Namespace Reference

8.5.1 Detailed Description

Module for handling random graphs.

Chapter 9

Python Webgraph Generator Class Documentation

9.1 pygel::BaseElements::AbstractEdge::AbstractEdge Class Reference

9.1.1 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.

9.2 pygel::Graph::AbstractGraph::AbstractGraph Class Reference

9.2.1 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.

9.3 pygel::BaseElements::AbstractVertex::AbstractVertex Class Reference

9.3.1 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.

9.4 pygel::RandomGraphs::ChooseEdges::ChooseEdges Class Reference

9.4.1 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](#)
- [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.

9.4.2 Member Function Documentation

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

Constructs a selector thread.

See also:

`RandomGraphs::PowerLawRandomGraph`

9.4.2.2 `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

9.4.3 Member Data Documentation

9.4.3.1 [pygel::RandomGraphs::ChooseEdges::ChooseEdges::endVertX](#)

See also:

`RandomGraphs::PowerLawRandomGraph`

9.4.3.2 [pygel::RandomGraphs::ChooseEdges::ChooseEdges::endVertY](#)

See also:

`RandomGraphs::PowerLawRandomGraph`

9.4.3.3 [pygel::RandomGraphs::ChooseEdges::ChooseEdges::noOfEdges](#)

See also:

`RandomGraphs::PowerLawRandomGraph`

9.4.3.4 [pygel::RandomGraphs::ChooseEdges::ChooseEdges::probA](#)**See also:**

RandomGraphs::PowerLawRandomGraph

9.4.3.5 [pygel::RandomGraphs::ChooseEdges::ChooseEdges::probB](#)**See also:**

RandomGraphs::PowerLawRandomGraph

9.4.3.6 [pygel::RandomGraphs::ChooseEdges::ChooseEdges::probC](#)**See also:**

RandomGraphs::PowerLawRandomGraph

9.4.3.7 [pygel::RandomGraphs::ChooseEdges::ChooseEdges::probD](#)**See also:**

RandomGraphs::PowerLawRandomGraph

9.4.3.8 list [pygel::RandomGraphs::ChooseEdges::ChooseEdges::serialEdgeList = \[\]](#) [static]

Common serial edge list.

Updated by each thread in a semaphoric operation

9.4.3.9 [pygel::RandomGraphs::ChooseEdges::ChooseEdges::startVertX](#)**See also:**

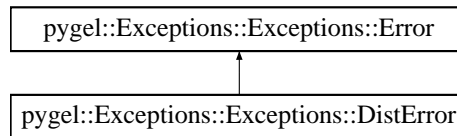
RandomGraphs::PowerLawRandomGraph

9.4.3.10 [pygel::RandomGraphs::ChooseEdges::ChooseEdges::startVertY](#)**See also:**

RandomGraphs::PowerLawRandomGraph

9.5 pygel::Exceptions::Exceptions::DistError Class Reference

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



9.5.1 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

9.5.2 Member Function Documentation

9.5.2.1 def pygel::Exceptions::Exceptions::DistError::__init__(self, message)

Constructs a [DistError](#) exception.

Parameters:

message Error message

9.6 pygel::BaseElements::Edge::Edge Class Reference

9.6.1 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.

9.6.2 Member Function Documentation

9.6.2.1 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

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

Get the end vertex.

Returns:

endVertex End vertex of type BaseElements::Vertex

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

Get the start vertex.

Returns:

startVertex Start vertex of type BaseElements::Vertex

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

Set the end vertex.

Parameters:

endVertex End vertex of type BaseElements::Vertex

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

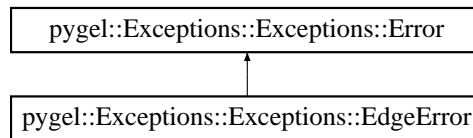
Set the start vertex.

Parameters:

startVertex Start vertex of type BaseElements::Vertex

9.7 pygel::Exceptions::Exceptions::EdgeError Class Reference

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



9.7.1 Detailed Description

Represents a [EdgeError](#) exception.

It handles different types of graph edge related exceptions

Public Member Functions

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

Public Attributes

- `edgeNumber`
Vertex number for which the exception occurred.
- `message`
Error message

9.7.2 Member Function Documentation

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

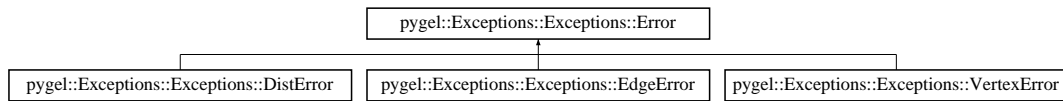
Constructs a [EdgeError](#) exception.

Parameters:

- edgeNumber* Edge number for which the exception occurred
message Error message

9.8 pygel::Exceptions::Exceptions::Error Class Reference

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



9.8.1 Detailed Description

Empty base class from which all exceptions are derived.

9.9 pygel::Exceptions::Exceptions::ErrorMessages Class Reference

9.9.1 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 number already exists'
- string **edgeNotFound** = 'Edge number not found'
- string **distAddOne** = 'Probabilities do not add to one'

9.10 pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph Class Reference

9.10.1 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 [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 [getInDegreeDistribution](#)
Get in-degree distribution.
- def [getOutDegreeDistribution](#)

Get out-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 [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.

9.10.2 Member Function Documentation

9.10.2.1 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::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

9.10.2.2 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::addVertex (self, vertexNumber)`

Adds a vertex.

Should be used with care

Parameters:

vertexNumber Vertex number of vertex to be added

Exceptions:

PackageExceptions::VertexError

9.10.2.3 def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::deleteEdge (*self*, *edgeNumber*)

Delete an edge.

Parameters:

edgeNumber Edge number to be deleted

9.10.2.4 def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::deleteVertex (*self*, *vertexNumber*)

Deletes a vertex.

Should be used with care

Parameters:

vertexNumber Vertex number to be deleted

9.10.2.5 def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::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

9.10.2.6 def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::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

9.10.2.7 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::getEdges (self)`

Get all graph edges.

Returns:

edgeIndex Dictionary of edges, indexed by edge number

9.10.2.8 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::getInDegree-Distribution (self)`

Get in-degree distribution.

Returns:

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

9.10.2.9 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::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

9.10.2.10 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::getNumberOfIn-Neighbors (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

9.10.2.11 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::getNumberOfOut-Neighbors (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

9.10.2.12 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::getOutDegree-Distribution (self)`

Get out-degree distribution.

Returns:

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

9.10.2.13 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::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

9.10.2.14 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::getVertices (self)`

Get all graph vertices.

Returns:

vertexIndex Dictionary of vertices, indexed by vertex number

9.10.2.15 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::getVerticesByIn-Degree (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

9.10.2.16 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::getVerticesByOut-Degree (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

9.10.2.17 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::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

9.10.2.18 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::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

9.10.2.19 `def pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph::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

9.11 pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph Class Reference

9.11.1 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 [PowerLawRandomGraph::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. Probability of choosing quadrant A.
- [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**

9.11.2 Member Function Documentation

9.11.2.1 **def pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph::__init__ (*self*, *size*, *noOfEdges*)**

Constructs an empty graph.

Parameters:

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

9.11.2.2 **def pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph::generate (*self*, *noOfThreads*)**

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

9.11.2.3 **def pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph::populate (*self*)**

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

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

9.11.2.4 **def pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph::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*

9.11 `pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph` Class Reference

9.11.2.5 `def pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph::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

9.11.3 Member Data Documentation

9.11.3.1 `pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph::probA`

Parameters of the RMat algorithm. Probability of choosing quadrant A.

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

Todo

Add description about choosing these probabilities

9.11.3.2 `pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph::serialEdgeList`

Temporary storage of edges.

Maintained for achieving performance

9.12 pygel::BaseElements::Vertex::Vertex Class Reference

9.12.1 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.

9.12.2 Member Function Documentation

9.12.2.1 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

9.12.2.2 def pygel::BaseElements::Vertex::Vertex::getVertexNumber (*self*)

Get vertex number.

Returns:

`vertexNumber` `Vertex` number of this vertex

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

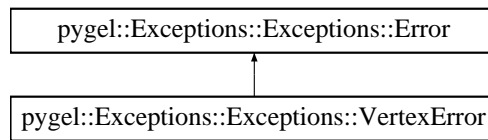
Set vertex number.

Parameters:

vertexNumber New vertex number

9.13 pygel::Exceptions::Exceptions::VertexError Class Reference

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



9.13.1 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

9.13.2 Member Function Documentation

9.13.2.1 `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

9.14 pygel::BaseElements::WeightedVertex::WeightedVertex Class Reference

9.14.1 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.

9.14.2 Member Function Documentation

9.14.2.1 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

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

Get vertex weight.

Returns:

vertexWeight Vertex weight of this vertex

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

Set vertex weight.

Parameters:

vertexWeight New vertex weight

9.15 pygel::BaseElements::WeightedVertices::WeightedVertices Class Reference

9.15.1 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.

9.15.2 Member Function Documentation

9.15.2.1 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

9.15.2.2 `def pygel::BaseElements::WeightedVertices::WeightedVertices::delVertex (self,
vertexNumber)`

Delete vertex from the collection.

Parameters:

vertexNumber Vertex number of the vertex to be deleted

9.15.2.3 `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

9.15.2.4 `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

9.15.2.5 `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

9.15.2.6 `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

Chapter 10

Python Webgraph Generator Page Documentation

10.1 General Todo List

Todo

Integrate with [psyco](#)

10.2 Todo List

Member [pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph::probA](#)

Add description about choosing these probabilities

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

Index

- `__init__`
 - `pygel::BaseElements::Edge::Edge`, 30
 - `pygel::BaseElements::Vertex::Vertex`, 44
 - `pygel::BaseElements::WeightedVertex::WeightedVertex`, 46
 - `pygel::Exceptions::Exceptions::DistError`, 29
 - `pygel::Exceptions::Exceptions::EdgeError`, 32
 - `pygel::Exceptions::Exceptions::VertexError`, 45
 - `pygel::RandomGraphs::ChooseEdges::ChooseEdges`, 27
 - `pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph`, 42
- `addEdge`
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 36
- `addVertex`
 - `pygel::BaseElements::WeightedVertices::WeightedVertices`, 48
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 36
- `BaseElements`, 17
- `Basic Elements`, 13
- `deleteEdge`
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 37
- `deleteVertex`
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 37
- `delVertex`
 - `pygel::BaseElements::WeightedVertices::WeightedVertices`, 48
- `endVertX`
 - `pygel::RandomGraphs::ChooseEdges::ChooseEdges`, 27
- `endVertY`
 - `pygel::RandomGraphs::ChooseEdges::ChooseEdges`, 27
- `Exceptions`, 18
- `findEdge`
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 37
- `findVertex`
 - `pygel::BaseElements::WeightedVertices::WeightedVertices`, 49
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 37
- `findWeight`
 - `pygel::BaseElements::WeightedVertices::WeightedVertices`, 49
- `generate`
 - `pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph`, 42
- `getEdges`
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 37
- `getEndVertex`
 - `pygel::BaseElements::Edge::Edge`, 30
- `getInDegreeDistribution`
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 38
- `getInNeighbors`
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 38
- `getNumberOfInNeighbors`
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 38
- `getNumberOfOutNeighbors`
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 38
- `getOutDegreeDistribution`
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 38
- `getOutNeighbors`
 - `pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph`, 39
- `getStartVertex`
 - `pygel::BaseElements::Edge::Edge`, 31
- `getVertexNumber`
 - `pygel::BaseElements::Vertex::Vertex`, 44
- `getVertices`
 - `pygel::BaseElements::WeightedVertices::WeightedVertices`, 49

- pygel::Graph::NumberedEdge-Graph::NumberedEdgeGraph, 39
- getVerticesByInDegree
 - pygel::Graph::NumberedEdge-Graph::NumberedEdgeGraph, 39
- getVerticesByOutDegree
 - pygel::Graph::NumberedEdge-Graph::NumberedEdgeGraph, 39
- getWeight
 - pygel::BaseElements::Weighted-Vertex::WeightedVertex, 46
- Graph, 15, 19
- hasVertex
 - pygel::BaseElements::Weighted-Vertices::WeightedVertices, 49
 - pygel::Graph::NumberedEdge-Graph::NumberedEdgeGraph, 39
- noOfEdges
 - pygel::RandomGraphs::Choose-Edges::ChooseEdges, 27
- Package Exceptions, 14
- PackageExceptions, 20
- populate
 - pygel::RandomGraphs::PowerLawRandom-Graph::PowerLawRandomGraph, 42
- probA
 - pygel::RandomGraphs::Choose-Edges::ChooseEdges, 27
 - pygel::RandomGraphs::PowerLawRandom-Graph::PowerLawRandomGraph, 43
- probB
 - pygel::RandomGraphs::Choose-Edges::ChooseEdges, 28
- probC
 - pygel::RandomGraphs::Choose-Edges::ChooseEdges, 28
- probD
 - pygel::RandomGraphs::Choose-Edges::ChooseEdges, 28
- pygel::BaseElements::AbstractEdge::AbstractEdge, 23
- pygel::BaseElements::AbstractVertex::AbstractVertex, 25
- pygel::BaseElements::Edge::Edge, 30
- pygel::BaseElements::Edge::Edge
 - __init__, 30
 - getEndVertex, 30
 - getStartVertex, 31
 - setEndVertex, 31
 - setStartVertex, 31
- pygel::BaseElements::Vertex::Vertex, 44
- pygel::BaseElements::Vertex::Vertex
 - __init__, 44
 - getVertexNumber, 44
 - setVertexNumber, 44
- pygel::BaseElements::WeightedVertex::WeightedVertex, 46
- pygel::BaseElements::WeightedVertex::Weighted-Vertex
 - __init__, 46
 - getWeight, 46
 - setWeight, 46
- pygel::BaseElements::WeightedVertices::WeightedVertices, 48
- pygel::BaseElements::WeightedVertices::Weighted-Vertices
 - addVertex, 48
 - delVertex, 48
 - findVertex, 49
 - findWeight, 49
 - getVertices, 49
 - hasVertex, 49
- pygel::Exceptions::Exceptions::DistError, 29
- pygel::Exceptions::Exceptions::DistError
 - __init__, 29
- pygel::Exceptions::Exceptions::EdgeError, 32
- pygel::Exceptions::Exceptions::EdgeError
 - __init__, 32
- pygel::Exceptions::Exceptions::Error, 33
- pygel::Exceptions::Exceptions::ErrorMessages, 34
- pygel::Exceptions::Exceptions::VertexError, 45
- pygel::Exceptions::Exceptions::VertexError
 - __init__, 45
- pygel::Graph::AbstractGraph::AbstractGraph, 24
- pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph, 35
- pygel::Graph::NumberedEdgeGraph::Numbered-EdgeGraph
 - addEdge, 36
 - addVertex, 36
 - deleteEdge, 37
 - deleteVertex, 37
 - findEdge, 37
 - findVertex, 37
 - getEdges, 37
 - getInDegreeDistribution, 38
 - getInNeighbors, 38
 - getNumberOfInNeighbors, 38
 - getNumberOfOutNeighbors, 38
 - getOutDegreeDistribution, 38
 - getOutNeighbors, 39
 - getVertices, 39
 - getVerticesByInDegree, 39
 - getVerticesByOutDegree, 39
 - hasVertex, 39

- readEdges, [40](#)
 - writeEdges, [40](#)
- pygel::RandomGraphs::ChooseEdges::ChooseEdges, [26](#)
- pygel::RandomGraphs::ChooseEdges::ChooseEdges
 - __init__, [27](#)
 - endVertX, [27](#)
 - endVertY, [27](#)
 - noOfEdges, [27](#)
 - probA, [27](#)
 - probB, [28](#)
 - probC, [28](#)
 - probD, [28](#)
 - selectVertex, [27](#)
 - serialEdgeList, [28](#)
 - startVertX, [28](#)
 - startVertY, [28](#)
- pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph, [41](#)
- pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph
 - __init__, [42](#)
 - generate, [42](#)
 - populate, [42](#)
 - probA, [43](#)
 - serialEdgeList, [43](#)
 - setProbs, [42](#)
 - writeEdges, [42](#)
- Random Graphs, [16](#)
- RandomGraphs, [21](#)
- readEdges
 - pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph, [40](#)
- selectVertex
 - pygel::RandomGraphs::ChooseEdges::ChooseEdges, [27](#)
- serialEdgeList
 - pygel::RandomGraphs::ChooseEdges::ChooseEdges, [28](#)
 - pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph, [43](#)
- setEndVertex
 - pygel::BaseElements::Edge::Edge, [31](#)
- setProbs
 - pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph, [42](#)
- setStartVertex
 - pygel::BaseElements::Edge::Edge, [31](#)
- setVertexNumber
 - pygel::BaseElements::Vertex::Vertex, [44](#)
- setWeight
 - pygel::BaseElements::WeightedVertex::WeightedVertex, [46](#)
- startVertX
 - pygel::RandomGraphs::ChooseEdges::ChooseEdges, [28](#)
- startVertY
 - pygel::RandomGraphs::ChooseEdges::ChooseEdges, [28](#)
- writeEdges
 - pygel::Graph::NumberedEdgeGraph::NumberedEdgeGraph, [40](#)
 - pygel::RandomGraphs::PowerLawRandomGraph::PowerLawRandomGraph, [42](#)