DirectedGraph Class Documentation

Author: Farcas Mihai-Cristian
Date: 03.04.2024

Introduction

This document serves as the documentation for the **DirectedGraph** class in Python.

Class Definition

DirectedGraph

Represents a directed graph.

Attributes:

- **num_vertices** (*int*): The number of vertices in the graph.
- **num_edges** (*int*): The number of edges in the graph.
- **inbound_edges** (*dict*): A dictionary mapping each vertex to its inbound edges.
- **outbound_edges** (*dict*): A dictionary mapping each vertex to its outbound edges.
- **edges** (*dict*): A dictionary mapping each edge (source, destination) to its cost.

Methods

__init__(self, num_vertices=0, num_edges=0)

Initializes a **DirectedGraph** instance.

- **Parameters:**
    - **num_vertices** (*int, optional*): The initial number of vertices (default is 0).
    - **num_edges** (*int, optional*): The initial number of edges (default is 0).

add_vertex(self, vertex_id)

Adds a new vertex to the graph.

- **Parameters:**
    - **vertex_id** (*int*): The ID of the vertex to be added.

add_edge(self, source, destination, cost)

Adds a new edge to the graph.

- **Parameters:**
    - **source** (*int*): The source vertex of the edge.

    - **destination** (*int*): The destination vertex of the edge.
    - **cost** (*int*): The cost associated with the edge.

remove_edge(self, source, destination)

Removes an edge from the graph.

- **Parameters:**
    - **source** (*int*): The source vertex of the edge.
    - **destination** (*int*): The destination vertex of the edge.

remove_vertex(self, vertex_id)

Removes a vertex from the graph.

- **Parameters:**
    - **vertex_id** (*int*): The ID of the vertex to be removed.

get_num_vertices(self)

Returns the current number of vertices in the graph.

- **Returns:**
    - (*int*): The current number of vertices.

get_num_edges(self)

Returns the current number of edges in the graph.

- **Returns:**
    - (*int*): The current number of edges.

vertices_iter(self)

Returns an iterator over the vertices set.

- **Returns:**
    - (*iterator*): An iterator over the vertices.

outbound_edges_iter(self, vertex_id)

Returns an iterator over the outbound edges of the given vertex.

- **Parameters:**
    - **vertex_id** (*int*): The vertex for which the outbound edges are requested.
- **Returns:**
    - (*iterator*): An iterator over the outbound edges.

inbound_edges_iter(self, vertex_id)

Returns an iterator over the inbound edges of the given vertex.

- **Parameters:**
    - **vertex_id** (*int*): The vertex for which the inbound edges are requested.
- **Returns:**
    - (*iterator*): An iterator over the inbound edges.

edge_exists(self, source, destination)

Checks if the edge given by its source and destination exists.

- **Parameters:**
    - **source** (*int*): The source vertex of the edge.
    - **destination** (*int*): The destination vertex of the edge.
- **Returns:**
    - (*bool*): True if the edge exists, False otherwise.

get_edge_cost(self, edge)

Returns the cost of the given edge.

- **Parameters:**
    - **edge** (*tuple*): The edge specified by the user.
- **Returns:**
    - (*int*): The cost of the given edge.

set_edge_cost(self, edge, cost)

Sets the cost of a given edge.

- **Parameters:**
    - **edge** (*tuple*): The edge specified by the user.
    - **cost** (*int*): The cost of the given edge.

get_in_degree(self, vertex_id)

Returns the 'in' degree of the given vertex.

- **Parameters:**
    - **vertex_id** (*int*): The ID of the vertex.
- **Returns:**
    - (*int*): The 'in' degree of the given vertex.

get_out_degree(self, vertex_id)

Returns the 'out' degree of the given vertex.

- **Parameters:**

    - **vertex_id** (*int*): The ID of the vertex.
- **Returns:**
    - (*int*): The 'out' degree of the given vertex.

console_print(self)

Prints the graph's data in the console.

save_file(self, filename)

Saves the graph's data to a specified file.

- **Parameters:**
    - **filename** (*str*): The file in which the data is being saved.

load_file(self, filename)

Loads the graph's data from a given file.

- **Parameters:**
    - **filename** (*str*): The file from which the data is taken.

copy(self)

Returns a copy of the graph.

- **Returns:**
    - (*DirectedGraph*): A copy of the graph.