

dijkstra

Generated by Doxygen 1.9.6

1 Class Index	1
1.1 Class List	1
2 File Index	3
2.1 File List	3
3 Class Documentation	5
3.1 krawedz Struct Reference	5
3.1.1 Detailed Description	5
4 File Documentation	7
4.1 funkcje.h File Reference	7
4.1.1 Detailed Description	8
4.1.2 Function Documentation	8
4.1.2.1 dodaj_krawedz()	8
4.1.2.2 odczytaj_graf()	9
4.1.2.3 szukanie_tras()	9
4.1.2.4 wierzchowce()	10
4.1.2.5 wypisz_trase()	10
4.1.2.6 znajdz_najblizszy_wierzcholek()	11
4.2 funkcje.h	11
Index	13

Chapter 1

Class Index

1.1 Class List

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

krawedz	5
-----------------------------------	---

Chapter 2

File Index

2.1 File List

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

funkcje.h	7
-------------------------------------	---

Chapter 3

Class Documentation

3.1 krawedz Struct Reference

```
#include <funkcje.h>
```

Public Attributes

- int **nr_wierz_poczatkowy**
- int **nr_wierz_koncowy**
- double **waga**

3.1.1 Detailed Description

Tworzona struktura o nazwie "krawedz" tworzone sa zmienne o wartosciach parametrow krawedzi : z ktorego do ktorego wierzchołka i jego waga

The documentation for this struct was generated from the following file:

- [funkcje.h](#)

Chapter 4

File Documentation

4.1 funkcje.h File Reference

```
#include <iostream>
#include <random>
#include <chrono>
#include <fstream>
#include <vector>
#include <map>
#include <cmath>
#include <set>
#include <string>
#include <sstream>
#include <iomanip>
#include <format>
```

Classes

- struct [krawedz](#)

Functions

- bool [odczytaj_graf](#) (std::string nazwa_pliku, std::vector< [krawedz](#) > *krawedzie, std::set< int > *wierzcholki)
Funkcja co sprawdza i odczytuje dane z pliku krawedzie i wierzcholki.
- void [dodaj_krawedz](#) (int poczatek, int koniec, double waga, std::vector< [krawedz](#) > *krawedzie)
Funkcja do tworzenia listy krawedzi z jej wartosciami do kazdego parametra.
- bool [wierzchowce](#) (std::string nazwa_pliku, std::set< int > *wierzcholki)
Funkcja co sprawdza i odczytuje wierzcholki z pliku wierzcholki.
- std::string [szukanie_tras](#) (std::set< int > *zbior_wierzcholkow, std::vector< [krawedz](#) > *zbior_krawedzi, int poczatkowy_wierzcholek)

Funkcja do odszukiwania szukanych najkrotszych tras. Pobiera wierzcholki razem z ich dystansem bez powtorzen za pomoca "insert_or_assign". Dodawanie elementów odbywa się przy użyciu metody insert(), do której podajemy klucz do dodania (wierzcholek, INFINITY). Jeśli klucz już istnieje w zbiorze, operacja dodawania nie zmienia w żaden sposób zawartości już pobranych wierzchołkow. Następnie tworzenie zbiorow sprawdzonych wierzchołkow; wyluskanie wartosci z pod wzkaznika na wierzcholki, robienie kopii do usuwania jej; dopoki kolejka pobranych wierzchołkow nie jest pusta, to idzie przypisywanie do naimniejszego wierzchowka z funkcji "znajdz_najblizszy_↔ wierzcholek" ktory to wierzcholek jest razem z jestgo dystansem, jezeli juz wszystko zostalo przeczytane to konczy sie proces przypisywania; po nim do sprawdzonych wierzchołkow beda sie przepisywali wierzcholki bez dublowania, a z kolejki wierzchołkow beda usuwac sie sprawdzone wierzcholki; w kolejnym kroku wykonuje sie relaksacja; i na konie wypisywanie wszystkich informacji w postaci stringu.

- `std::string wypisz_trase` (`std::map< int, int > &poprzedniki`, `int &wierzcholek`)

Funkcka do wypisywania trasy od zadanego wierzcholka do wierzcholka o najmniejszej wadze, przy waronku, ze poprzedni wierzcholek nie jest 0.

- `int znajdz_najblizszy_wierzcholek` (`std::set< int > &kolejka_wierzchołkow`, `std::map< int, double > &dystans`)

Funkcja do odnalezienia najblizszego wierzcholka. Pszypisaanie początkowych wartosci do dystansu(INFINITY), a do pierwszego wierzcholka (0). Dopoki wierzcholki sa odczytywane z kolejki, idzie porownanie kazdego dystansu, aby odszukać najmniejszy z nich. Koniecznie jest odnaleziony wierzcholek o najkrotszym dystansie.

4.1.1 Detailed Description

4.1.2 Function Documentation

4.1.2.1 dodaj_krawedz()

```
void dodaj_krawedz (
    int poczatek,
    int koniec,
    double waga,
    std::vector< krawedz > * krawedzie )
```

Funkcja do tworzenia listy krawedzi z jej watrosciami do kazdego parametra.

dodajemy nową krawedź do wektora

Parameters

<i>poczatek</i>	wierzcholek oznaczający początek krawedzi
<i>koniec</i>	wierzcholek oznaczający koniec krawedzi
<i>waga</i>	waga samej krawedzi między początkowym a końcowym wierzchołkiem
<i>krawedzie</i>	wskaznik na wektor przechowujący krawędzi
<i>poczatek</i>	wierzcholek początku
<i>koniec</i>	wierzcholek końca
<i>waga</i>	waga od początku do końca
<i>krawedzie</i>	wskaznik na wektor krawędzi

Funkcja do tworzenia listy krawędzi z jej wartościami do każdego parametra.

Parameters

<i>poczatek</i>	wierzchołek początku
<i>koniec</i>	wierzchołek końca
<i>waga</i>	waga od początku do końca
<i>krawedzie</i>	wskaźnik na wektor krawędzi

4.1.2.2 odczytaj_graf()

```
bool odczytaj_graf (
    std::string nazwa_pliku,
    std::vector< krawedz > * krawedzie,
    std::set< int > * wierzcholki )
```

Funkcja co sprawdza i odczytuje dane z pliku krawedzie i wierzcholki.

Parameters

<i>nazwa_pliku</i>	nazwa pliku z którego wczytujemy graf
<i>krawedzie</i>	wskaźnik na wektor przechowujący krawędzi
<i>wierzcholki</i>	wskaźnik na zbior przechowujący wierzcholki

Returns

zwraca czy funkcja się udała

4.1.2.3 szukanie_tras()

```
std::string szukanie_tras (
    std::set< int > * zbior_wierzcholkow,
    std::vector< krawedz > * zbior_krawedzi,
    int poczatkowy_wierzcholek )
```

Funkcja do odszukiwania szukanych najkrótszych tras. Pobiera wierzcholki razem z ich dystansem bez powtórzeń za pomocą "insert_or_assign". Dodawanie elementów odbywa się przy użyciu metody insert(), do której podajemy klucz do dodania (wierzchołek, INFINITY). Jeśli klucz już istnieje w zbiorze, operacja dodawania nie zmienia w żaden sposób zawartości już pobranych wierzchołków. Następnie tworzenie zbiorów sprawdzonych wierzchołków; wyluskanie wartości z pod wskaźnika na wierzcholki, robienie kopii do usuwania jej; dopóki kolejka pobranych wierzchołków nie jest pusta, to idzie przypisywanie do najmniejszego wierzchołka z funkcji "znajdz_najblizszy_wierzcholek" który to wierzchołek jest razem z jego dystansem, jeżeli już wszystko zostało przeczytane to kończy się proces przypisywania; po nim do sprawdzonych wierzchołków będą się przepisywali wierzcholki bez dublowania, a z kolejki wierzchołków będą usuwane sprawdzone wierzcholki; w kolejnym kroku wykonuje się relaksacja; i na koniec wypisywanie wszystkich informacji w postaci stringu.

Parameters

<i>zbior_wierzchołkow</i>	przypisanie zbioru wierzchołkow
<i>zbior_krawedzi</i>	przypisanie zbioru krawedzi
<i>początkowy_wierzcholek</i>	początkowy wierzcholek

Returns

zwraca szukane trasy w postaci string

4.1.2.4 wierzchowce()

```
bool wierzchowce (
    std::string nazwa_pliku,
    std::set< int > * wierzcholki )
```

Funkcja co sprawdza i odczytuje wierzcholki z pliku wierzcholki.

Parameters

<i>nazwa_pliku</i>	nazwa pliku z którego wczytujemy wierzcholki
<i>wierzcholki</i>	wskaznik na zbior przechowujący wierzcholki

Returns

zwraca czy funkcja się udała

4.1.2.5 wypisz_trase()

```
std::string wypisz_trase (
    std::map< int, int > & poprzedniki,
    int & wierzcholek )
```

Funkcja do wypisywania trasy od zadanego wierzchołka do wierzchołka o najmniejszej wadze, przy warunku, że poprzedni wierzchołek nie jest 0.

Parameters

<i>poprzedniki</i>	zbior poprzedników wierzchołków
<i>wierzcholek</i>	wierzchołek do którego jest odszukiwana trasa

Returns

zwraca trasę z końcowym wierzchołkiem razem z jego poprzednikiem

4.1.2.6 znajdz_najblizszy_wierzcholek()

```
int znajdz_najblizszy_wierzcholek (
    std::set< int > & kolejka_wierzchołkow,
    std::map< int, double > & dystans )
```

Funkcja do odnalezienia najbliższego wierzchołka. Przypisanie początkowych wartości do dystansu(INFINITY), a do pierwszego wierzchołka (0). Dopóki wierzchołki są odczytywane z kolejki, idzie porównanie każdego dystansu, aby odszukać najmniejszy z nich. Koniecznie jest odnaleziony wierzchołek o najkrótszym dystansie.

Parameters

<i>kolejka_wierzchołkow</i>	rozpatrywane wierzchołki
<i>dystans</i>	dystans do wierzchołkow

Returns

zwraca najmniejszy wierzchołek

Funkcja do odnalezienia najbliższego wierzchołka. Przypisanie początkowych wartości do dystansu(INFINITY), a do pierwszego wierzchołka (0). Dopóki wierzchołki są odczytywane z kolejki, idzie porównanie każdego dystansu, aby odszukać najmniejszy z nich. Koniecznie jest odnaleziony wierzchołek o najkrótszym dystansie

4.2 funkcje.h

[Go to the documentation of this file.](#)

```
00001 #include <iostream>
00002 #include <random>
00003 #include <chrono>
00004 #include <fstream>
00005 #include <vector>
00006 #include <map>
00007 #include <cmath>
00008 #include <set>
00009 #include <string>
00010 #include <sstream>
00011 #include <iomanip>
00012 #include <format>
00016 //Deklaracje (Nagłówki) Funkcji
00018 #ifndef FUNKCJE_H // zabezpieczenie przed wielokrotnym dołączaniem pliku nagłówkowego
00019 #define FUNKCJE_H
00020
00028 struct krawedz
00029 {
00030     int nr_wierz_początkowy;
00031     int nr_wierz_koncowy;
00032     double waga;
00033 };
00034
00043 bool odczytaj_graf(std::string nazwa_pliku, std::vector<krawedz>* krawedzie, std::set<int>*
    wierzchołki);
00044
00045
00055 void dodaj_krawedz(int początek, int koniec, double waga, std::vector<krawedz>* krawedzie);
00056
00064 bool wierzchowce(std::string nazwa_pliku, std::set<int>* wierzchołki);
00065
00066
00067
00076 void dodaj_krawedz(int początek, int koniec, double waga, std::vector<krawedz>* krawedzie);
00077
```

```
00078
00079
00100 std::string szukanie_tras(std::set<int>* zbior_wierzchołkow, std::vector<krawedz>* zbior_krawedzi, int
    poczatkowy_wierzcholek);
00101
00102
00112 std::string wypisz_trase(std::map<int, int>& poprzedniki, int& wierzcholek);
00113
00127 int znajdz_najblizszy_wierzcholek(std::set<int>& kolejka_wierzchołkow, std::map<int, double>&
    dystans);
00128
00129 #endif
00130
00131
```


Index

dodaj_krawedz
funkcje.h, [8](#)

funkcje.h, [7](#)
dodaj_krawedz, [8](#)
odczytaj_graf, [9](#)
szukanie_tras, [9](#)
wierzchowce, [10](#)
wypisz_trase, [10](#)
znajdz_najblizszy_wierzcholek, [11](#)

krawedz, [5](#)

odczytaj_graf
funkcje.h, [9](#)

szukanie_tras
funkcje.h, [9](#)

wierzchowce
funkcje.h, [10](#)

wypisz_trase
funkcje.h, [10](#)

znajdz_najblizszy_wierzcholek
funkcje.h, [11](#)