Poznanie osoby x

0.1

Wygenerowano przez Doxygen 1.8.9.1

Cz, 11 cze 2015 07:49:07

Spis treści

1 program - poznanie osoby x **1**

2 Indeks klas **1**

2.1 Lista klas 1

3 Indeks plików **2**

3.1 Lista plików 2

4 Dokumentacja klas **2**

4.1 Dokumentacja klasy CEdge 2

4.1.1 Opis szczegółowy 3

4.1.2 Dokumentacja przyjaciół i funkcji związanych 3

4.1.3 Dokumentacja atrybutów składowych 3

4.2 Dokumentacja klasy CGraph 3

4.2.1 Opis szczegółowy 4

4.2.2 Dokumentacja konstruktora i destruktora 4

4.2.3 Dokumentacja funkcji składowych 4

4.2.4 Dokumentacja atrybutów składowych 6

4.3 Dokumentacja klasy CNode 6

4.3.1 Opis szczegółowy 7

4.3.2 Dokumentacja przyjaciół i funkcji związanych 7

4.3.3 Dokumentacja atrybutów składowych 7

4.4 Dokumentacja szablonu klasy Lista< typ > 7

4.4.1 Opis szczegółowy 7

4.4.2 Dokumentacja konstruktora i destruktora 8

4.4.3 Dokumentacja funkcji składowych 8

4.4.4 Dokumentacja atrybutów składowych 8

4.5 Dokumentacja klasy queue 9

4.5.1 Opis szczegółowy 9

4.5.2 Dokumentacja konstruktora i destruktora 9

4.5.3 Dokumentacja funkcji składowych 9

4.5.4 Dokumentacja atrybutów składowych 10

4.6 Dokumentacja klasy queue_node 10

4.6.1 Opis szczegółowy 10

4.6.2 Dokumentacja atrybutów składowych 10

5 Dokumentacja plików **11**

5.1 Dokumentacja pliku format.h 11

5.1.1 Opis szczegółowy 11

5.1.2 Dokumentacja funkcji 11

5.2 Dokumentacja pliku global.h . . . 12
5.2.1 Opis szczegółowy . . . 12
5.2.2 Dokumentacja definicji . . . 12
5.2.3 Dokumentacja funkcji . . . 13
5.2.4 Dokumentacja zmiennych . . . 13
5.3 Dokumentacja pliku graph.cpp . . . 14
5.4 Dokumentacja pliku graph.hh . . . 15
5.5 Dokumentacja pliku main.cpp . . . 15
5.5.1 Dokumentacja funkcji . . . 15
5.6 Dokumentacja pliku queue.cpp . . . 16
5.7 Dokumentacja pliku queue.hh . . . 16
5.8 Dokumentacja pliku stack.hh . . . 16

Indeks **17**

1 program - poznanie osoby x

Autor

W. Makuch, W.Maluszynski

Data

Wersja

0.1

Program zrealizowany jest z wykorzystaniem funkcji winApi W pierwszym polu nalezy wpisac plik zawieracjacy konfiguracje grafu, nastepnie wczytac W drugim polu nalezy wpisac identyfikator pierwszej osoby, nastepnie wczytac W trzecim polu nalezy wpisac identyfikator drugiej osoby, nastepnie wczytac

program wyswietla okineko ze znaleziona sciezka W przypadku blednej nazwy pliku lub nieznalezienia sciezki program zglasza blad

2 Indeks klas

2.1 Lista klas

Tutaj znajdują się klasy, struktury, unie i interfejsy wraz z ich krótkimi opisami:

CEdge
Definicja klasy CEdge definuje krawedz grafu nie zawiera wag **2**

CGraph
Definicja klasy CGraph definije graf skierowany bez wagowaych krawedzi dziedzicy po klasie CBenchmark **3**

CNode
Definicja klasy CNode definiuje pojedynczy wezel grafu **6**

Lista< typ >
Definicja klasy Lista Przechowuje obiekt oraz wskaznik na nastepny i pole rozmiar. Zbudowana na szablonie 7

queue
Definicja klasy queue definicja kolejki ADT, kolejka typu FIFO zimplementowaa na liscie 9

queue_node
Definicja klasy queue_node definicja wezla dla kolejki definiuje pojedynczy element bedacy w kolejkce 10

3 Indeks plików

3.1 Lista plików

Tutaj znajduje się lista wszystkich plików z ich krótkimi opisami:

format.h 11

global.h 12

graph.cpp
Implementuje zdefiniowana klase grafu 14

graph.hh
Zwiera definicje klas CNode, CEdge, CGrpah CNode - wezel grafu CEdge - krawedz grafu CGraph - graf 15

main.cpp 15

queue.cpp
Implementuje zdefiniowana klase kolejki 16

queue.hh
Zawiera definicje klas queue_node, queue queue_node - wezel kolejki queue - kolejka 16

stack.hh
Definicja struktury danych Lista 16

4 Dokumentacja klas

4.1 Dokumentacja klasy CEdge

definicja klasy CEdge definuje krawedz grafu nie zawiera wag

`#include <graph.hh>`

Atrybuty prywatne

- CNode * prev
- CNode * next

Przyjaciele

- class CGraph

4.1.1 Opis szczegółowy

Definicja w linii 29 pliku graph.hh.

4.1.2 Dokumentacja przyjaciół i funkcji związanych

4.1.2.1 friend class CGraph [friend]

Definicja w linii 30 pliku graph.hh.

4.1.3 Dokumentacja atrybutów składowych

4.1.3.1 CNode* CEdge::next [private]

wskaznik na poprzedni wezel

Definicja w linii 32 pliku graph.hh.

4.1.3.2 CNode* CEdge::prev [private]

Definicja w linii 31 pliku graph.hh.

Dokumentacja dla tej klasy została wygenerowana z pliku:

- graph.hh

4.2 Dokumentacja klasy CGraph

definicja klasy CGraph definije graf skierowany bez wagowaych krawedzi dziedzicy po klasie CBenchmark

`#include <graph.hh>`

Metody publiczne

- CGraph (int v, int e)

  definicja konstruktora parametrycznego alokuje pamiec dla tablic wezlow krawdzi oraz dla macierzy sasiedztwa
- ~CGraph ()

  definicja destruktora
- void set_graph (int vertice1, int vertice2)

  definicja metody set_graph wiaze wezly wskaznikami krawdziom ustawia odpowiednie wezly wypelnia macierz sasiedztwa
- void print_matrix () const

  definicja metody print_maatrix wyswietla macierz sasiedztwa
- void DFS (int v)

  definicja metody DFSing przeszukuje graf algorytmem DFS (przeszukiwanie w glab) zaimplementowana rekuencyjnie wykorzystana w metodzie DFS wyposarzonej w timery
- void BFS (int v)

  definicja metody DFS wywoluje metode DFSing posiada timery zapisujace zmierzony czas do pliku
- void save_matrix (std::fstream &file) const

  definicja metody save_matrix zapsiuje macierz saciedztwa do pliku
- bool BFSPath (int start, int meta, Lista< int > *L)

  definicja metody BFSPath Oblicza najszybsza sciezke w grafie miedzy punktem startowym, a koncowym i zapisuje ja na stosie

Atrybuty publiczne

- int V
- int E
- int ∗∗ matrix
- bool ∗ visited

4.2.1 Opis szczegółowy

Definicja w linii 42 pliku graph.hh.

4.2.2 Dokumentacja konstruktora i destruktora

4.2.2.1 CGraph::CGraph (int *v*, int *e*)

lista odwiedzonych elementow grafu

Parametry

v	ilosc wezlow
e	ilosc krawedzi

Definicja w linii 25 pliku graph.cpp.

4.2.2.2 CGraph::∼CGraph ( )

Definicja w linii 41 pliku graph.cpp.

4.2.3 Dokumentacja funkcji składowych

4.2.3.1 void CGraph::BFS (int *v*)

Parametry

v	element poczatkowy od ktorego algorytm DFS rozpoczyna przeszukiwanie

Definicja w linii 80 pliku graph.cpp.

4.2.3.2 bool CGraph::BFSPath (int *start*, int *meta*, Lista< int > ∗ *L*)

Parametry

start	punkt startowy
meta	punkt koncowy
∗*L*	wskaznik do stosu, na ktorym sa zapisane elementy

Definicja w linii 109 pliku graph.cpp.

4.2.3.3 void CGraph::DFS (int *v*)

Parametry

v	element poczatkowy od ktorego algorytm rozpoczyna przeszukiwanie

Definicja w linii 68 pliku graph.cpp.

4.2.3.4 void CGraph::print_matrix ( ) const

Definicja w linii 58 pliku graph.cpp.

4.2.3.5 void CGraph::save_matrix (std::fstream & *file*) const

Parametry

file	uchwyt do pliku

Definicja w linii 15 pliku graph.cpp.

4.2.3.6 void CGraph::set_graph (int *vertice1*, int *vertice2*)

Parametry

edge1	krawdz do ktorej przypisujemy poszczegolne wierzcholki
vertice1	przypisywany pierwszy wierzcholek
vertice2	przypisywany drugi wierzcholek

Definicja w linii 52 pliku graph.cpp.

4.2.4 Dokumentacja atrybutów składowych

4.2.4.1 int CGraph::E

ilosc wezlow

Definicja w linii 45 pliku graph.hh.

4.2.4.2 int** CGraph::matrix

ilosc krawedzi

Definicja w linii 47 pliku graph.hh.

4.2.4.3 int CGraph::V

Definicja w linii 44 pliku graph.hh.

4.2.4.4 bool* CGraph::visited

macierz sasiedztwa

Definicja w linii 48 pliku graph.hh.

Dokumentacja dla tej klasy została wygenerowana z plików:

- graph.hh
- graph.cpp

4.3 Dokumentacja klasy CNode

definicja klasy CNode definiuje pojedynczy wezel grafu

`#include <graph.hh>`

Atrybuty prywatne

- int value

Przyjaciele

- class CEdge
- class CGraph

4.3.1 Opis szczegółowy

Definicja w linii 18 pliku graph.hh.

4.3.2 Dokumentacja przyjaciół i funkcji związanych

4.3.2.1 friend class CEdge [friend]

Definicja w linii 19 pliku graph.hh.

4.3.2.2 friend class CGraph [friend]

Definicja w linii 20 pliku graph.hh.

4.3.3 Dokumentacja atrybutów składowych

4.3.3.1 int CNode::value [private]

Definicja w linii 21 pliku graph.hh.

Dokumentacja dla tej klasy została wygenerowana z pliku:

- graph.hh

4.4 Dokumentacja szablonu klasy Lista< typ >

definicja klasy Lista Przechowuje obiekt oraz wskaznik na nastepny i pole rozmiar. Zbudowana na szablonie.

`#include <stack.hh>`

Metody publiczne

- Lista ()
  definicja konstruktora bezparametrycznego Zeruje rozmiar, ustawia wskaznik na NULL.
- ~Lista ()
  definicja destruktora Zeruje rozmiar, Kasuje wszystkie obiektry/elementy.
- void push (typ element)
  definicja metody push
- typ pop ()
  definicja metody pop
- int size () const
  deinicja metody size
- void wyswietl ()

Atrybuty publiczne

- Lista< typ > * nastepny
- typ dane
- int rozmiar

4.4.1 Opis szczegółowy

template<typename typ>class Lista< typ >

Definicja w linii 16 pliku stack.hh.

4.4.2 Dokumentacja konstruktora i destruktora

4.4.2.1 template<typename typ > Lista< typ >::Lista ( )

ilosc elementow/obiektow

Definicja w linii 36 pliku stack.hh.

4.4.2.2 template<typename typ > Lista< typ >::~Lista ( )

Definicja w linii 97 pliku stack.hh.

4.4.3 Dokumentacja funkcji składowych

4.4.3.1 template<typename typ > typ Lista< typ >::pop ( )

Zwraca

usuwany element Ustawia wskaznik na poprzedni element zwraca i kasuje ostatni element.
0 i wyswietla komunikat gdy lista jest pusta.

Definicja w linii 65 pliku stack.hh.

4.4.3.2 template<typename typ > void Lista< typ >::push (typ *element*)

Parametry

[element]	dodawany element na koniec listy Zwieksza rozmiar, alokuje pamiec, przypisuje element do pola klasy.

Definicja w linii 48 pliku stack.hh.

4.4.3.3 template<typename typ > int Lista< typ >::size ( ) const

Zwraca

ilosc elementow przechowywanych na liscie.

Definicja w linii 86 pliku stack.hh.

4.4.3.4 template<typename typ > void Lista< typ >::wyswietl ( )

Definicja w linii 105 pliku stack.hh.

4.4.4 Dokumentacja atrybutów składowych

4.4.4.1 template<typename typ> typ Lista< typ >::dane

wskaznik na nastepny obiekt/element

Definicja w linii 19 pliku stack.hh.

4.4.4.2 template<typename typ> Lista<typ>* Lista< typ >::nastepny

Definicja w linii 18 pliku stack.hh.

4.4.4.3 template<typename typ> int Lista< typ >::rozmiar

przechowywana informacja/obiekt/element

Definicja w linii 20 pliku stack.hh.

Dokumentacja dla tej klasy została wygenerowana z pliku:

- stack.hh

4.5 Dokumentacja klasy queue

definicja klasy queue definicja kolejki ADT, kolejka typu FIFO zimplementowaa na liscie

`#include <queue.hh>`

Metody publiczne

- bool is_empty () const
- void push (int element)

  definicja metody push dodaje element na koniec kolejki
- void pop ()

  definicja metody pop usuwa element z poczatku kolejki
- ∼queue ()

  definicja destruktora
- queue ()

  desfinicja konstruktora bezparametrycznego ustawia wskazniki na NULL
- void print () const

  definicja metody print wyswietla zawartosc kolejki
- int front (void)

Atrybuty publiczne

- queue_node ∗ first
- queue_node ∗ last

4.5.1 Opis szczegółowy

Definicja w linii 29 pliku queue.hh.

4.5.2 Dokumentacja konstruktora i destruktora

4.5.2.1 queue::∼queue ( )

Definicja w linii 49 pliku queue.cpp.

4.5.2.2 queue::queue ( ) [inline]

Definicja w linii 57 pliku queue.hh.

4.5.3 Dokumentacja funkcji składowych

4.5.3.1 int queue::front (void)

Definicja w linii 63 pliku queue.cpp.

4.5.3.2 bool queue::is_empty ( ) const

wskaznik na ostatni element

Definicja w linii 55 pliku queue.cpp.

4.5.3.3 void queue::pop ( )

Definicja w linii 23 pliku queue.cpp.

4.5.3.4 void queue::print ( ) const

Definicja w linii 35 pliku queue.cpp.

4.5.3.5 void queue::push (int *element*)

Parametry

element	dodawany element

Definicja w linii 9 pliku queue.cpp.

4.5.4 Dokumentacja atrybutów składowych

4.5.4.1 queue_node* queue::first

Definicja w linii 31 pliku queue.hh.

4.5.4.2 queue_node* queue::last

wskaznik na pierwszy element

Definicja w linii 32 pliku queue.hh.

Dokumentacja dla tej klasy została wygenerowana z plików:

- queue.hh
- queue.cpp

4.6 Dokumentacja klasy queue_node

definicja klasy queue_node definicja wezla dla kolejki definiuje pojedynczy element bedacy w kolejkce

`#include <queue.hh>`

Atrybuty publiczne

- queue_node * next
- int data

4.6.1 Opis szczegółowy

Definicja w linii 17 pliku queue.hh.

4.6.2 Dokumentacja atrybutów składowych

4.6.2.1 int queue_node::data

wskaznik na nastepny

Definicja w linii 20 pliku queue.hh.

4.6.2.2 queue_node* queue_node::next

Definicja w linii 19 pliku queue.hh.

Dokumentacja dla tej klasy została wygenerowana z pliku:

- queue.hh

5 Dokumentacja plików

5.1 Dokumentacja pliku format.h

```
#include <string>
#include <windows.h>
```

Funkcje

- string intToStr (int n)

  definicja funkcji intToString konwersja ina na string
- LPSTR ZamienFormat (Lista< int > *C)

  definicja funkcji zamienia liste intow(sciezka w grafie) na LPSTR

5.1.1 Opis szczegółowy

przechowuje funkcje konwertujace string

5.1.2 Dokumentacja funkcji

5.1.2.1 string intToStr (int *n*)

Parametry

n	zadany int

Zwraca

zwracany string

Definicja w linii 20 pliku format.h.

5.1.2.2 LPSTR ZamienFormat (Lista< int > * *C*)

Parametry

*C	wskaznik na liste

Zwraca

sciezka w formacie LPSTR

Definicja w linii 45 pliku format.h.

5.2 Dokumentacja pliku global.h

```
#include <windows.h>
#include <tchar.h>
#include <iostream>
```

Definicje

- #define ID_BUTTON1 101
- #define ID_BUTTON2 102
- #define ID_BUTTON3 103
- #define ID_STATICTEXTBOX1 201
- #define ID_STATICTEXTBOX2 202
- #define ID_STATICTEXTBOX3 203
- #define ID_COMANDLINE 301

Funkcje

- LRESULT CALLBACK WindowProcedure (HWND, UINT, WPARAM, LPARAM)

Zmienne

- TCHAR szClassName [ ] = _T("WindowsApp")
- HWND hCommandLine
- HWND hExecButton
- HWND hStaticDebugBox
- HWND hCommandLine2
- HWND hExecButton2
- HWND hStaticDebugBox2
- HWND hCommandLine3
- HWND hExecButton3
- HWND hStaticDebugBox3
- LPSTR CommandLineBuf
- fstream plik
- Lista< int > C
- int i
- int j
- int licznik =0
- int start_path
- int end_path
- CGraph * G
- LPSTR tmp_string
- HWND hwnd

5.2.1 Opis szczegółowy

przechowuje zmienne globalne

5.2.2 Dokumentacja definicji

5.2.2.1 #define ID_BUTTON1 101

Definicja w linii 22 pliku global.h.

5.2.2.2 #define ID_BUTTON2 102

Definicja w linii 23 pliku global.h.

5.2.2.3 #define ID_BUTTON3 103

Definicja w linii 24 pliku global.h.

5.2.2.4 #define ID_COMANDLINE 301

Definicja w linii 28 pliku global.h.

5.2.2.5 #define ID_STATICTEXTBOX1 201

Definicja w linii 25 pliku global.h.

5.2.2.6 #define ID_STATICTEXTBOX2 202

Definicja w linii 26 pliku global.h.

5.2.2.7 #define ID_STATICTEXTBOX3 203

Definicja w linii 27 pliku global.h.

5.2.3 Dokumentacja funkcji

5.2.3.1 LRESULT CALLBACK WindowProcedure (HWND , UINT , WPARAM , LPARAM)

Definicja w linii 128 pliku main.cpp.

5.2.4 Dokumentacja zmiennych

5.2.4.1 Lista<int> C

Definicja w linii 55 pliku global.h.

5.2.4.2 LPSTR CommandLineBuf

Definicja w linii 49 pliku global.h.

5.2.4.3 int end_path

Definicja w linii 59 pliku global.h.

5.2.4.4 CGraph* G

Definicja w linii 60 pliku global.h.

5.2.4.5 HWND hCommandLine

Definicja w linii 37 pliku global.h.

5.2.4.6 HWND hCommandLine2

Definicja w linii 40 pliku global.h.

5.2.4.7 HWND hCommandLine3

Definicja w linii 43 pliku global.h.

5.2.4.8 HWND hExecButton

Definicja w linii 38 pliku global.h.

5.2.4.9 HWND hExecButton2

Definicja w linii 41 pliku global.h.

5.2.4.10 HWND hExecButton3

Definicja w linii 44 pliku global.h.

5.2.4.11 HWND hStaticDebugBox

Definicja w linii 39 pliku global.h.

5.2.4.12 HWND hStaticDebugBox2

Definicja w linii 42 pliku global.h.

5.2.4.13 HWND hStaticDebugBox3

Definicja w linii 45 pliku global.h.

5.2.4.14 HWND hwnd

Definicja w linii 63 pliku global.h.

5.2.4.15 int i

Definicja w linii 56 pliku global.h.

5.2.4.16 int j

Definicja w linii 56 pliku global.h.

5.2.4.17 int licznik =0

Definicja w linii 57 pliku global.h.

5.2.4.18 fstream plik

Definicja w linii 54 pliku global.h.

5.2.4.19 int start_path

Definicja w linii 58 pliku global.h.

5.2.4.20 TCHAR szClassName[ ] = _T("WindowsApp")

Definicja w linii 34 pliku global.h.

5.2.4.21 LPSTR tmp_string

Definicja w linii 61 pliku global.h.

5.3 Dokumentacja pliku graph.cpp

implementuje zdefiniowana klase grafu

```
#include "graph.hh"
#include "queue.hh"
#include "stack.hh"
#include <iostream>
#include <fstream>
#include <windows.h>
```

5.4 Dokumentacja pliku graph.hh

zwiera definicje klas CNode, CEdge, CGrpah CNode - wezel grafu CEdge - krawedz grafu CGraph - graf

```
#include <fstream>
#include "stack.hh"
```

Komponenty

- class CNode

  definicja klasy CNode definiuje pojedynczy wezel grafu
- class CEdge

  definicja klasy CEdge definuje krawedz grafu nie zawiera wag
- class CGraph

  definicja klasy CGraph definije graf skierowany bez wagowaych krawedzi dziedzicy po klasie CBenchmark

5.5 Dokumentacja pliku main.cpp

```
#include <iostream>
#include <string>
#include <fstream>
#include <sstream>
#include "graph.hh"
#include "queue.hh"
#include "stack.hh"
#include "format.h"
#include "global.h"
```

Funkcje

- int WINAPI WinMain (HINSTANCE hThisInstance, HINSTANCE hPrevInstance, LPSTR lpszArgument, int nCmdShow)
- LRESULT CALLBACK WindowProcedure (HWND hwnd, UINT message, WPARAM wParam, LPARAM l↩ Param)

5.5.1 Dokumentacja funkcji

5.5.1.1 LRESULT CALLBACK WindowProcedure (HWND *hwnd,* UINT *message,* WPARAM *wParam,* LPARAM *lParam*)

Definicja w linii 128 pliku main.cpp.

5.5.1.2 int WINAPI WinMain (HINSTANCE *hThisInstance,* HINSTANCE *hPrevInstance,* LPSTR *lpszArgument,* int *nCmdShow*)

Definicja w linii 32 pliku main.cpp.

5.6 Dokumentacja pliku queue.cpp

implementuje zdefiniowana klase kolejki

```
#include "queue.hh"
#include <iostream>
```

5.7 Dokumentacja pliku queue.hh

zawiera definicje klas queue_node, queue queue_node - wezel kolejki queue - kolejka

```
#include <iostream>
```

Komponenty

- class queue_node

  definicja klasy queue_node definicja wezla dla kolejki definiuje pojedynczy element bedacy w kolejkce
- class queue

  definicja klasy queue definicja kolejki ADT, kolejka typu FIFO zimplementowaa na liscie

5.8 Dokumentacja pliku stack.hh

definicja struktury danych Lista

```
#include <iostream>
```

Komponenty

- class Lista< typ >

  definicja klasy Lista Przechowuje obiekt oraz wskaznik na nastepny i pole rozmiar. Zbudowana na szablonie.

Skorowidz

~CGraph
CGraph, 4
~Lista
Lista, 8
~queue
queue, 9

BFS
CGraph, 4
BFSPath
CGraph, 4

C
global.h, 13
CEdge, 2
CGraph, 3
CNode, 7
next, 3
prev, 3
CGraph, 3
~CGraph, 4
BFS, 4
BFSPath, 4
CEdge, 3
CGraph, 4
CNode, 7
DFS, 4
E, 6
matrix, 6
print_matrix, 4
save_matrix, 4
set_graph, 6
V, 6
visited, 6
CNode, 6
CEdge, 7
CGraph, 7
value, 7
CommandLineBuf
global.h, 13

DFS
CGraph, 4
dane
Lista, 8
data
queue_node, 10

E
CGraph, 6
end_path
global.h, 13

first
queue, 10
format.h, 11
intToStr, 11
ZamienFormat, 11
front
queue, 9

G
global.h, 13
global.h, 12
C, 13
CommandLineBuf, 13
end_path, 13
G, 13
hCommandLine, 13
hCommandLine2, 13
hCommandLine3, 13
hExecButton, 13
hExecButton2, 14
hExecButton3, 14
hStaticDebugBox, 14
hStaticDebugBox2, 14
hStaticDebugBox3, 14
hwnd, 14
i, 14
ID_BUTTON1, 12
ID_BUTTON2, 12
ID_BUTTON3, 13
ID_COMANDLINE, 13
ID_STATICTEXTBOX1, 13
ID_STATICTEXTBOX2, 13
ID_STATICTEXTBOX3, 13
j, 14
licznik, 14
plik, 14
start_path, 14
szClassName, 14
tmp_string, 14
WindowProcedure, 13
graph.cpp, 14
graph.hh, 15

hCommandLine
global.h, 13
hCommandLine2
global.h, 13
hCommandLine3
global.h, 13
hExecButton
global.h, 13
hExecButton2
global.h, 14
hExecButton3
global.h, 14
hStaticDebugBox
global.h, 14
hStaticDebugBox2
global.h, 14

hStaticDebugBox3
global.h, 14
hwnd
global.h, 14

i
global.h, 14
ID_BUTTON1
global.h, 12
ID_BUTTON2
global.h, 12
ID_BUTTON3
global.h, 13
ID_COMANDLINE
global.h, 13
ID_STATICTEXTBOX1
global.h, 13
ID_STATICTEXTBOX2
global.h, 13
ID_STATICTEXTBOX3
global.h, 13
intToStr
format.h, 11
is_empty
queue, 9

j
global.h, 14

last
queue, 10
licznik
global.h, 14
Lista
~Lista, 8
dane, 8
Lista, 8
nastepny, 8
pop, 8
push, 8
rozmiar, 8
size, 8
wyswietl, 8
Lista< typ >, 7

main.cpp, 15
WinMain, 15
WindowProcedure, 15
matrix
CGraph, 6

nastepny
Lista, 8
next
CEdge, 3
queue_node, 10

plik
global.h, 14
pop
Lista, 8
queue, 9
prev
CEdge, 3
print
queue, 10
print_matrix
CGraph, 4
push
Lista, 8
queue, 10

queue, 9
~queue, 9
first, 10
front, 9
is_empty, 9
last, 10
pop, 9
print, 10
push, 10
queue, 9
queue.cpp, 16
queue.hh, 16
queue_node, 10
data, 10
next, 10

rozmiar
Lista, 8

save_matrix
CGraph, 4
set_graph
CGraph, 6
size
Lista, 8
stack.hh, 16
start_path
global.h, 14
szClassName
global.h, 14

tmp_string
global.h, 14

V
CGraph, 6
value
CNode, 7
visited
CGraph, 6

WinMain
main.cpp, 15
WindowProcedure
global.h, 13
main.cpp, 15
wyswietl
Lista, 8

ZamienFormat
format.h, 11