ADT

0.1

Generated by Doxygen 1.8.6

Mon May 23 2016 17:27:49

Contents

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

Chapter 2

Class Index

2.1 Class List

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

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

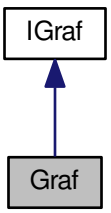
Chapter 4

Class Documentation

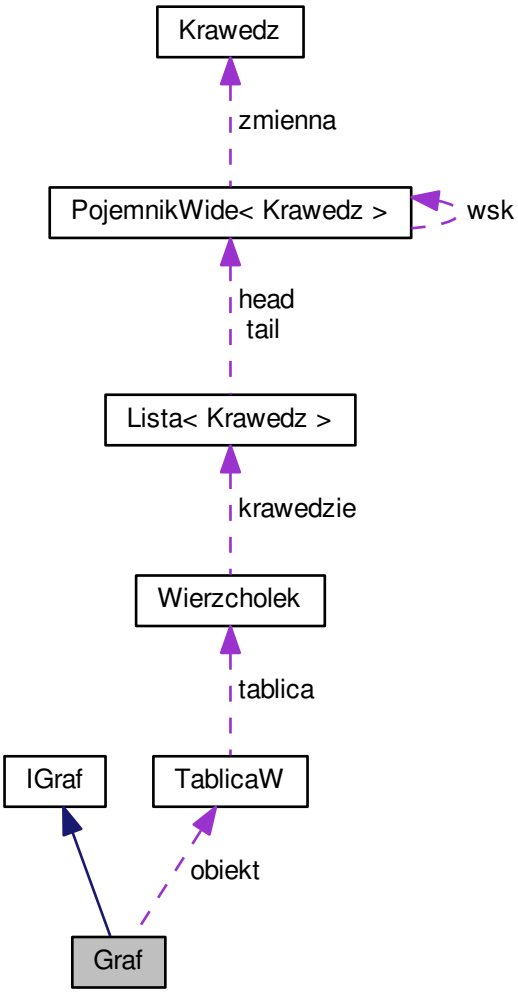
4.1 Graf Class Reference

`#include <Graf.hh>`

Inheritance diagram for Graf:



Collaboration diagram for Graf:



Public Member Functions

- void DodajW (string pozycja)

  Dodawanie wierzcholka do grafu.
- bool DodajK (string poczatek, string koniec, int waga=1)

  Dodawanie krawedzi do grafu.
- bool UsunK (string poczatek, string koniec)
- bool UsunW (string pozycja)
- Wierzcholek & WezW (int indeks)
- int WyszukajW (string pozycja)
- void Wyswietl ()

Private Attributes

- TablicaW obiekt

4.1.1 Detailed Description

Definition at line 15 of file Graf.hh.

4.1.2 Member Function Documentation

4.1.2.1 bool Graf::DodajK (string *poczatek,* string *koniec,* int *waga* = 1) [virtual]

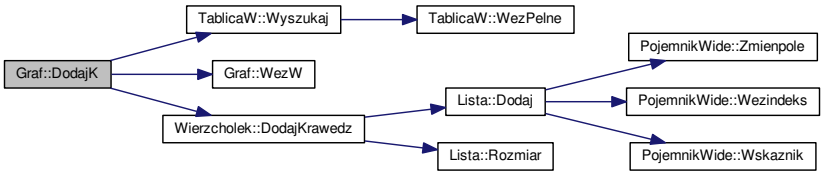
Parameters

in	*poczatek*	jeden z dwoch wierzcholkow, ktore laczy krawedz
in	*poczatek*	jeden z dwoch wierzcholkow, ktore laczy krawedz
in	*waga*	waga krawedzi

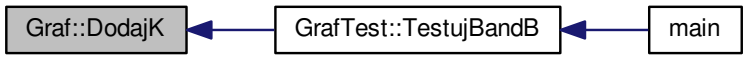
Implements IGraf.

Definition at line 22 of file Graf.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.2.2 void Graf::DodajW (string *pozycja*) [virtual]

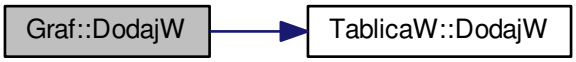
Parameters

in	*pozycja*	okresla nazwe identyfikujaca dany wierzcholek

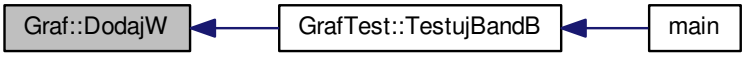
Implements IGraf.

Definition at line 9 of file Graf.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



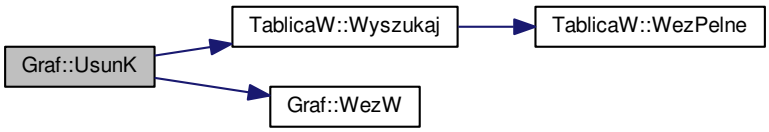
4.1.2.3 bool Graf::UsunK (string *poczatek*, string *koniec*) [virtual]

Usuwanie krawedzi z grafu

Implements IGraf.

Definition at line 45 of file Graf.cpp.

Here is the call graph for this function:



4.1.2.4 bool Graf::UsunW (string *pozycja*) [virtual]

Usuwa podany wierzcholek i przylegajace do niego krawedzie

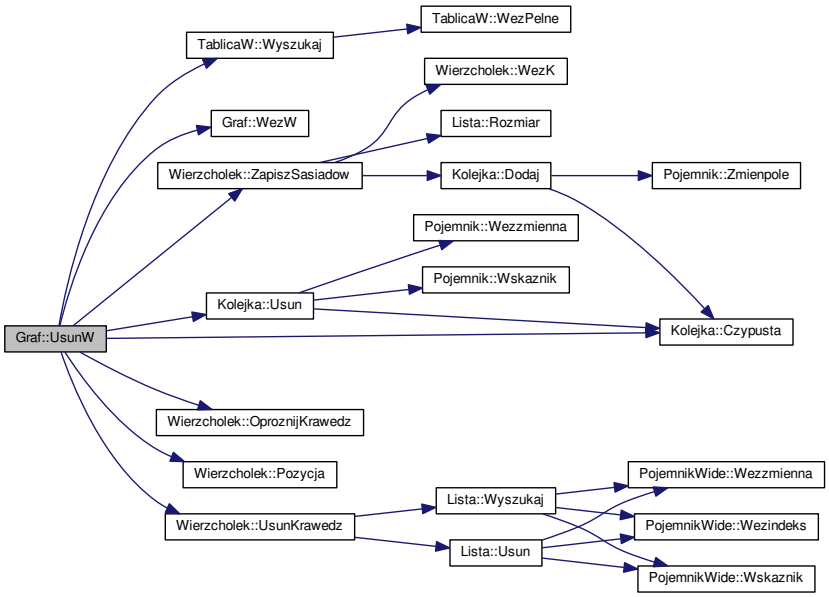
Parameters

in	*pozycja*	identyfikator wierzcholka do usuniecia

Implements IGraf.

Definition at line 75 of file Graf.cpp.

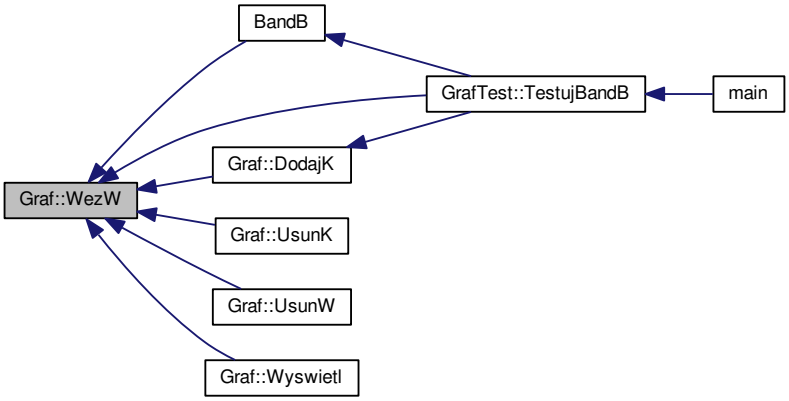
Here is the call graph for this function:



4.1.2.5 Wierzcholek& Graf::WezW (int *indeks*) [inline]

Definition at line 24 of file Graf.hh.

Here is the caller graph for this function:

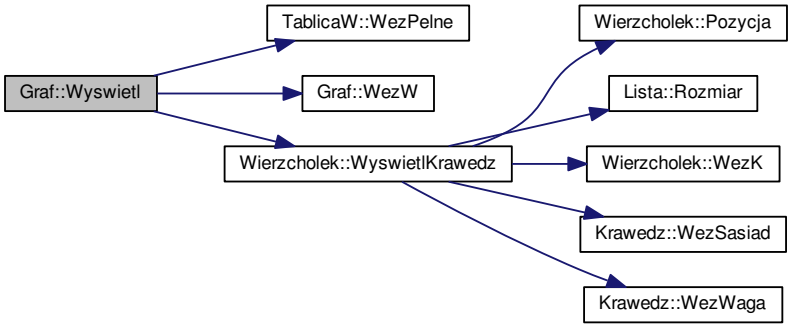


4.1.2.6 void Graf::Wyswietl ( )

Wyswietla Wszystkie wierzcholki i przylegajace do nich krawedzie

Definition at line 97 of file Graf.cpp.

Here is the call graph for this function:



4.1.2.7 int Graf::WyszukajW (string *pozycja*)

Wyszukuje dany wierzcholek grafu na podstawie jego identyfikatora

Parameters

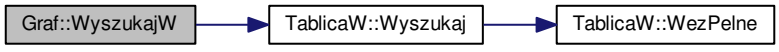
in	*pozycja*	identyfikator wierzcholka

Return values

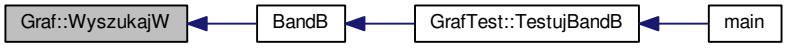
indeks	indeks, pod ktorym przechowywany jest dany wierzcholek w grafie

Definition at line 111 of file Graf.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.3 Member Data Documentation

4.1.3.1 TablicaW Graf::obiekt [private]

Definition at line 16 of file Graf.hh.

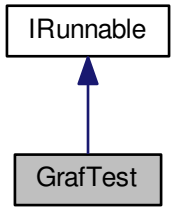
The documentation for this class was generated from the following files:

- Graf.hh
- Graf.cpp

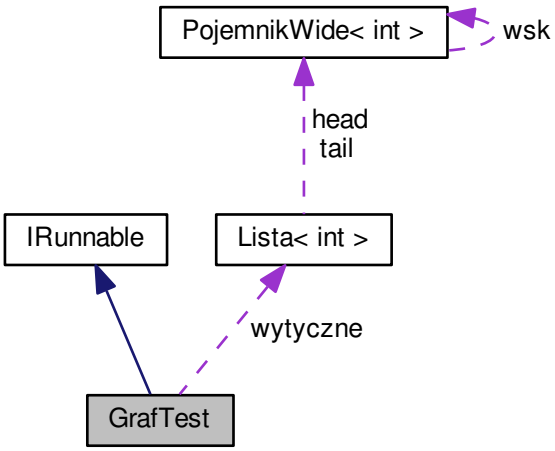
4.2 GrafTest Class Reference

`#include <GrafTest.hh>`

Inheritance diagram for GrafTest:



Collaboration diagram for GrafTest:



Public Member Functions

- bool Przygotuj (string NazwaPlikuWytyczne, string pNazwaPlikuDane)
- void TestujBandB ()

  Testowanie algorytmu BandB.
- ~GrafTest ()

Private Attributes

- Lista< int > wytyczne
- string NazwaPlikuDane

4.2.1 Detailed Description

Definition at line 17 of file GrafTest.hh.

4.2.2 Constructor & Destructor Documentation

4.2.2.1 GrafTest::~GrafTest ( ) [inline]

Definition at line 25 of file GrafTest.hh.

4.2.3 Member Function Documentation

4.2.3.1 bool GrafTest::Przygotuj (string *NazwaPlikuWytyczne*, string *pNazwaPlikuDane*) [virtual]

Odczytuje jaka ilosc wierzcholkow ma byc zapisana w grafie i skladuje odczytane wartosci na liscie. Wartosci odczytywane sa z pliku. Przykladowa zawartosc pliku: 10 100 1000. Metoda zapisuje takze nazwe pliku ,ktory zawiera nazwy wierzcholkow

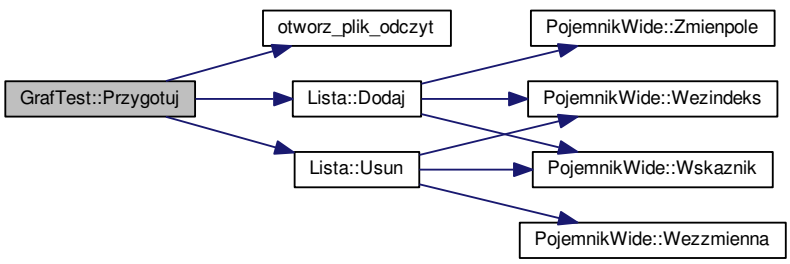
Parameters

in	*NazwaPliku-Wytyczne*	nazwa pliku, w ktorum przechowywane sa dane na temat ilosci wierzcholkow do zapisania w grafie
in	*pNazwaPliku-Dane*	nazwa pliku, ktory przechowuje nazwy wierzcholkow

Implements IRunnable.

Definition at line 62 of file GrafTest.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

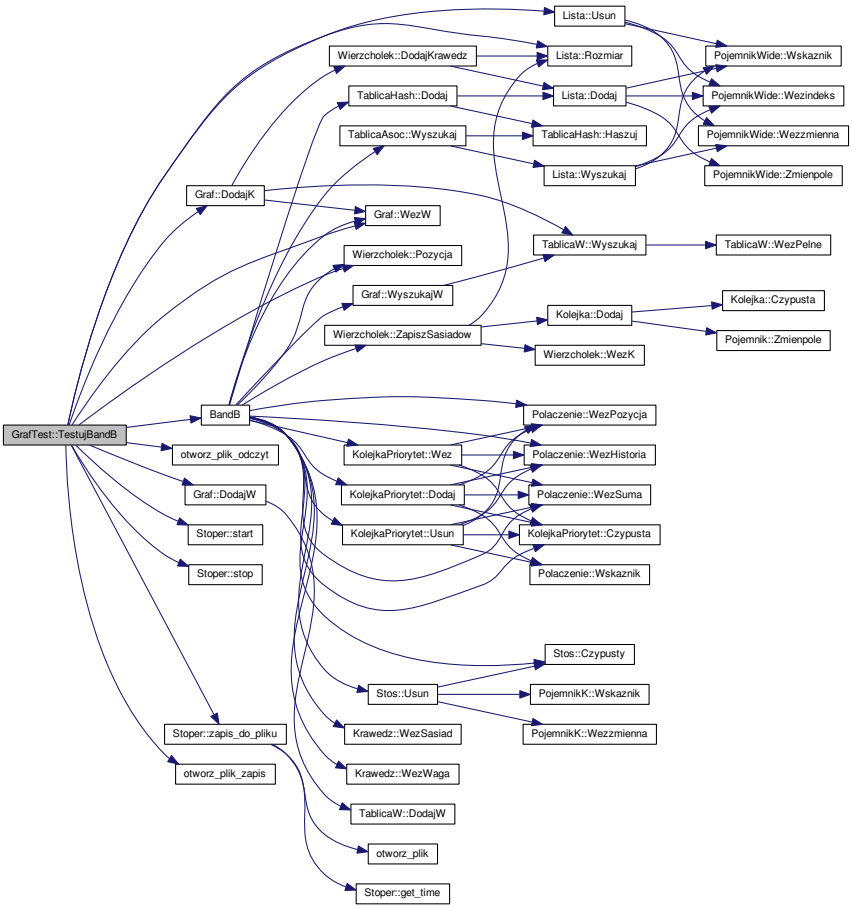
4.2.3.2 void GrafTest::TestujBandB ( ) [virtual]

Zapisuje odpowiednia ilosc elementow w grafie (odpowiednia-przechowywana w liscie "wytyczne". Krawedzi jest 2x wiecej niz wierzcholkow, aby zwiekszyc prawdopodobienstwo spojnosci grafu), wywoluje algorytm BandB dla stworzonego grafu, mierzy czas jego dzialania i zapisuje go do pliku "czasy.dat". Zapisuje takze w pliku "krawedzie.-dat" ilosc rozwinietych krawedzi przez algorytm BandB przy szukaniu najkrotszej drogi. Nazwy wierzcholkow grafu zczytywane sa z pliku o wczesniej zapisanej nazwie

Implements IRunnable.

Definition at line 88 of file GrafTest.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

4.2.4 Member Data Documentation

4.2.4.1 string GrafTest::NazwaPlikuDane [private]

Definition at line 19 of file GrafTest.hh.

4.2.4.2 Lista<int> GrafTest::wytyczne [private]

Definition at line 18 of file GrafTest.hh.

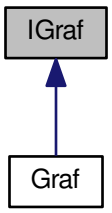
The documentation for this class was generated from the following files:

- GrafTest.hh
- GrafTest.cpp

4.3 IGraf Class Reference

`#include <IGraf.hh>`

Inheritance diagram for IGraf:



Public Member Functions

- virtual void DodajW (string pozycja)=0
- virtual bool DodajK (string poczatek, string koniec, int waga=1)=0
- virtual bool UsunK (string poczatek, string koniec)=0
- virtual bool UsunW (string pozycja)=0

4.3.1 Detailed Description

Definition at line 13 of file IGraf.hh.

4.3.2 Member Function Documentation

4.3.2.1 virtual bool IGraf::DodajK (string *poczatek,* string *koniec,* int *waga* = 1) [pure virtual]

Implemented in Graf.

4.3.2.2 virtual void IGraf::DodajW (string *pozycja*) [pure virtual]

Implemented in Graf.

4.3.2.3 virtual bool IGraf::UsunK (string *poczatek,* string *koniec*) [pure virtual]

Implemented in Graf.

4.3.2.4 virtual bool IGraf::UsunW (string *pozycja*) [pure virtual]

Implemented in Graf.

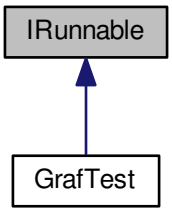
The documentation for this class was generated from the following file:

- IGraf.hh

4.4 IRunnable Class Reference

`#include <IRunnable.hh>`

Inheritance diagram for IRunnable:



Public Member Functions

- virtual bool Przygotuj (string NazwaPlikuWytyczne, string pNazwaPlikuDane)=0
- virtual void TestujBandB ()=0

4.4.1 Detailed Description

Definition at line 11 of file IRunnable.hh.

4.4.2 Member Function Documentation

4.4.2.1 virtual bool IRunnable::Przygotuj (string *NazwaPlikuWytyczne,* string *pNazwaPlikuDane*) [pure virtual]

Implemented in GrafTest.

4.4.2.2 virtual void IRunnable::TestujBandB ( ) [pure virtual]

Implemented in GrafTest.

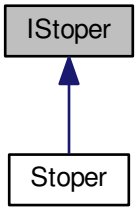
The documentation for this class was generated from the following file:

- IRunnable.hh

4.5 IStoper Class Reference

`#include <IStoper.hh>`

Inheritance diagram for IStoper:



Public Member Functions

- virtual void start ()=0
- virtual void stop ()=0
- virtual double get_time ()=0
- virtual bool zapis_do_pliku ()=0

4.5.1 Detailed Description

Definition at line 13 of file IStoper.hh.

4.5.2 Member Function Documentation

4.5.2.1 virtual double IStoper::get_time ( ) [pure virtual]

Implemented in Stoper.

4.5.2.2 virtual void IStoper::start ( ) [pure virtual]

Implemented in Stoper.

4.5.2.3 virtual void IStoper::stop ( ) [pure virtual]

Implemented in Stoper.

4.5.2.4 virtual bool IStoper::zapis_do_pliku ( ) [pure virtual]

Implemented in Stoper.

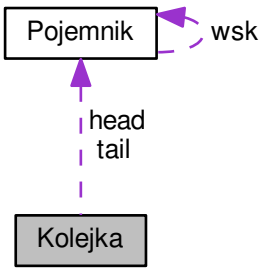
The documentation for this class was generated from the following file:

- IStoper.hh

4.6 Kolejka Class Reference

`#include <Kolejka.hh>`

Collaboration diagram for Kolejka:



Public Member Functions

- ~Kolejka ()
- void Dodaj (string elem)
- string Usun ()
- string Wez ()
- int Rozmiar ()
- bool Czypusta ()
- void Oproznij ()
- void Wyswietl ()

Private Attributes

- Pojemnik * head =NULL
- Pojemnik * tail =NULL
- int rozmiar =0

4.6.1 Detailed Description

Definition at line 13 of file Kolejka.hh.

4.6.2 Constructor & Destructor Documentation

4.6.2.1 Kolejka::~Kolejka ( ) [inline]

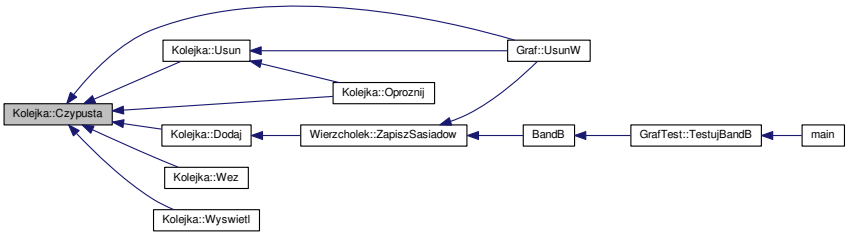
Definition at line 18 of file Kolejka.hh.

4.6.3 Member Function Documentation

4.6.3.1 bool Kolejka::Czypusta ( ) [inline]

Definition at line 23 of file Kolejka.hh.

Here is the caller graph for this function:



4.6.3.2 void Kolejka::Dodaj (string *elem*)

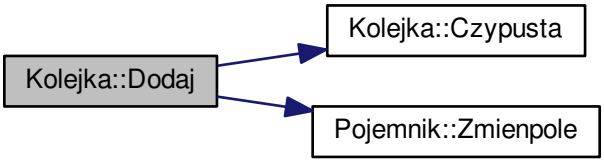
Dodaje element na koncu kolejki

Parameters

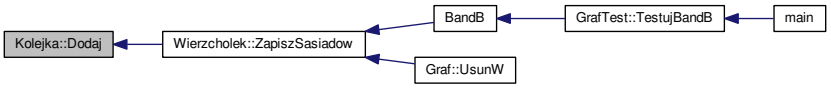
in	*elem*-	zmienna do przechowania

Definition at line 13 of file Kolejka.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

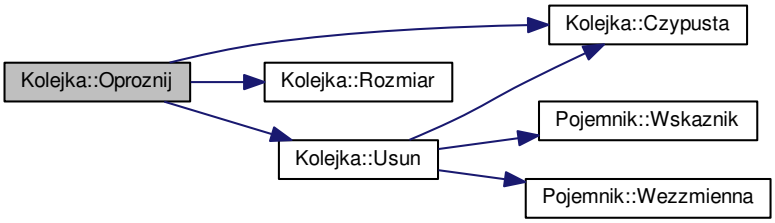


4.6.3.3 void Kolejka::Oproznij ( )

Usuwa wszystkie elementy kolejki

Definition at line 73 of file Kolejka.cpp.

Here is the call graph for this function:



4.6.3.4 int Kolejka::Rozmiar ( ) [inline]

Definition at line 22 of file Kolejka.hh.

Here is the caller graph for this function:

4.6.3.5 string Kolejka::Usun ( )

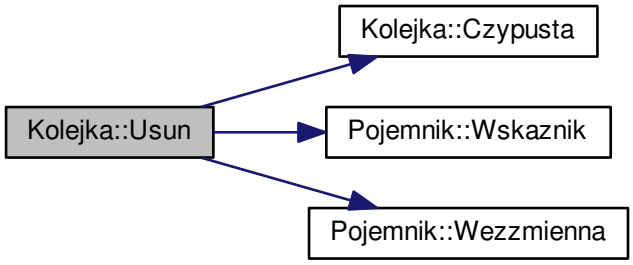
Usuwa element z poczatku kolejki i zwraca jego wartosc

Return values

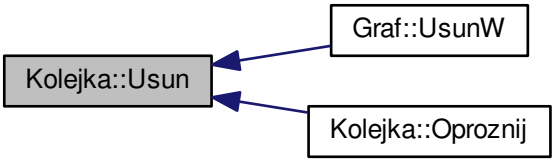
wartosc	usunietego elementu

Definition at line 37 of file Kolejka.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

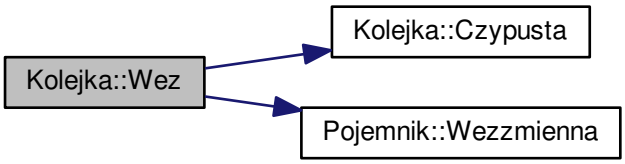


4.6.3.6 string Kolejka::Wez ( )

Zwraca wartosc pierwszego elementu w kolejce. Funkcja NIE sluzy do modyfikowania wartosci tego elementu

Definition at line 59 of file Kolejka.cpp.

Here is the call graph for this function:

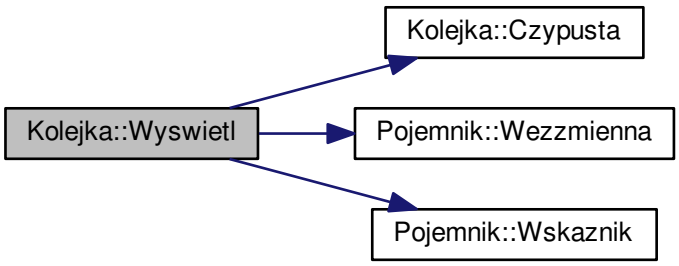


4.6.3.7 void Kolejka::Wyswietl ( )

Wyswietla wszystkie elementy kolejki od pierwszego do ostatniego

Definition at line 86 of file Kolejka.cpp.

Here is the call graph for this function:



4.6.4 Member Data Documentation

4.6.4.1 Pojemnik* Kolejka::head =NULL [private]

Definition at line 14 of file Kolejka.hh.

4.6.4.2 int Kolejka::rozmiar =0 [private]

Definition at line 16 of file Kolejka.hh.

4.6.4.3 Pojemnik* Kolejka::tail =NULL [private]

Definition at line 15 of file Kolejka.hh.

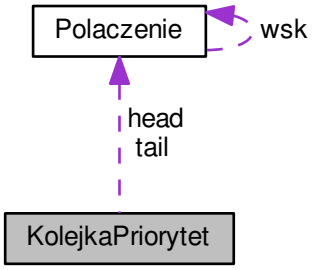
The documentation for this class was generated from the following files:

- Kolejka.hh
- Kolejka.cpp

4.7 KolejkaPriorytet Class Reference

`#include <KolejkaPriorytetowa.hh>`

Collaboration diagram for KolejkaPriorytet:



Public Member Functions

- ~KolejkaPriorytet ()
- void Dodaj (Polaczenie elem)
- Polaczenie Usun ()
- Polaczenie Wez ()
- int Rozmiar ()
- bool Czypusta ()
- void Oproznij ()

Private Attributes

- Polaczenie * head =NULL
- Polaczenie * tail =NULL
- int rozmiar =0

4.7.1 Detailed Description

Definition at line 13 of file KolejkaPriorytetowa.hh.

4.7.2 Constructor & Destructor Documentation

4.7.2.1 KolejkaPriorytet::~KolejkaPriorytet ( ) [inline]

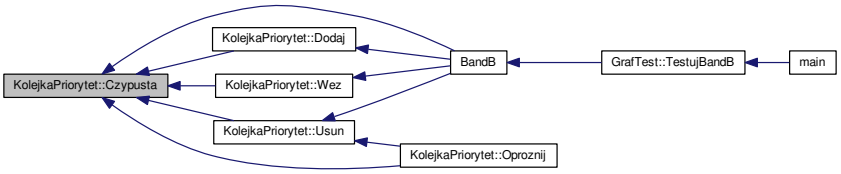
Definition at line 18 of file KolejkaPriorytetowa.hh.

4.7.3 Member Function Documentation

4.7.3.1 bool KolejkaPriorytet::Czypusta ( ) [inline]

Definition at line 23 of file KolejkaPriorytetowa.hh.

Here is the caller graph for this function:



4.7.3.2 void KolejkaPriorytet::Dodaj (Polaczenie *element*)

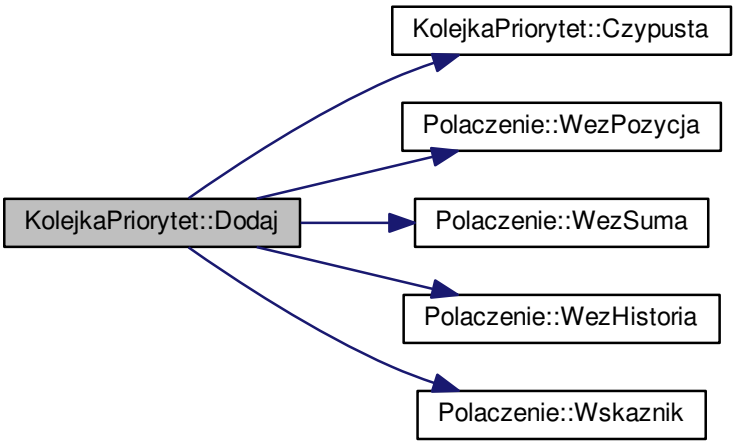
Element dodawany jest do kolejki w taki sposob, ze na jej poczatku zawsze skladowany jest element o najmniejszym parametrze "Suma"

Parameters

in	*element*	element, ktory ma byc przechowany w kolejce

Definition at line 14 of file KolejkaPriorytetowa.cpp.

Here is the call graph for this function:



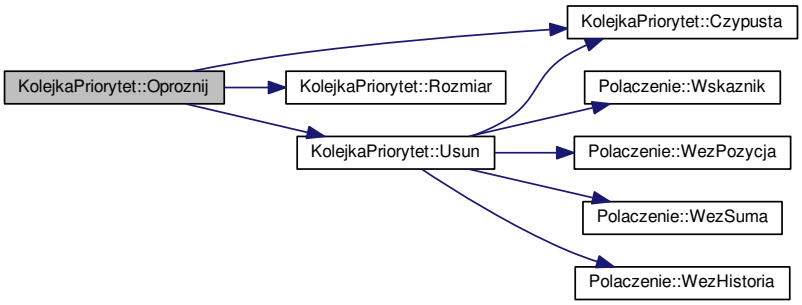
Here is the caller graph for this function:

4.7.3.3 void KolejkaPriorytet::Oproznij ( )

Usuwa wszystkie elementy kolejki priorytetowej

Definition at line 118 of file KolejkaPriorytetowa.cpp.

Here is the call graph for this function:



4.7.3.4 int KolejkaPriorytet::Rozmiar ( ) [inline]

Definition at line 22 of file KolejkaPriorytetowa.hh.

Here is the caller graph for this function:

4.7.3.5 Polaczenie KolejkaPriorytet::Usun ( )

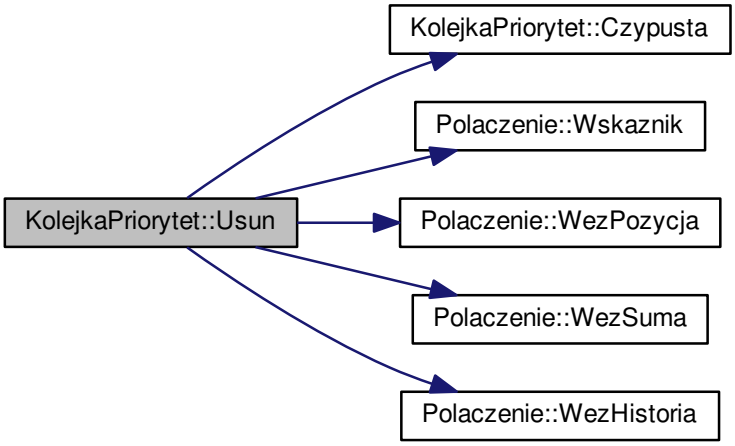
Usuwa element z poczatku kolejki (o najmniejszym parametrze "Suma") i zwraca go

Return values

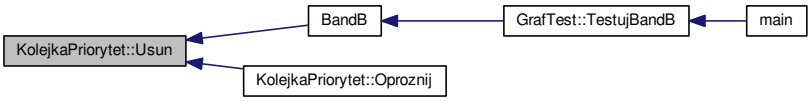
usuniety	element

Definition at line 78 of file KolejkaPriorytetowa.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

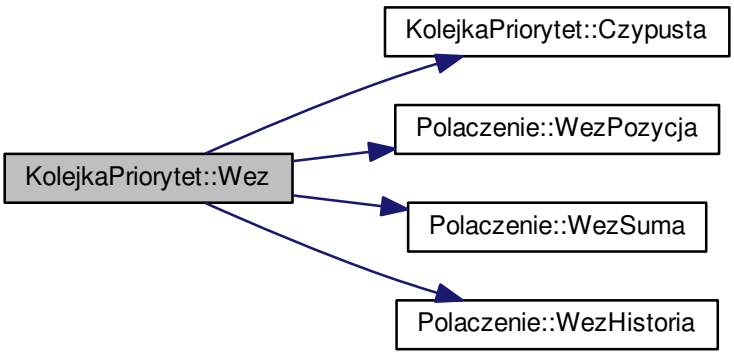


4.7.3.6 Polaczenie KolejkaPriorytet::Wez ( )

Zwraca wartosc pierwszego elementu w kolejce. Funkcja NIE sluzy do modyfikowania wartosci tego elementu

Definition at line 102 of file KolejkaPriorytetowa.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

4.7.4 Member Data Documentation

4.7.4.1 Polaczenie* KolejkaPriorytet::head =NULL [private]

Definition at line 14 of file KolejkaPriorytetowa.hh.

4.7.4.2 int KolejkaPriorytet::rozmiar =0 [private]

Definition at line 16 of file KolejkaPriorytetowa.hh.

4.7.4.3 Polaczenie* KolejkaPriorytet::tail =NULL [private]

Definition at line 15 of file KolejkaPriorytetowa.hh.

The documentation for this class was generated from the following files:

- KolejkaPriorytetowa.hh
- KolejkaPriorytetowa.cpp

4.8 Krawedz Class Reference

```
#include <Krawedz.hh>
```

Public Member Functions

- Krawedz ()
- Krawedz (string psasiad, int pwaga=1)
- string WezSasiad ()
- int WezWaga ()
- bool operator== (Krawedz druga)

Private Attributes

- string sasiad ="0"
- int waga =1

4.8.1 Detailed Description

Definition at line 13 of file Krawedz.hh.

4.8.2 Constructor & Destructor Documentation

4.8.2.1 Krawedz::Krawedz ( ) [inline]

Definition at line 19 of file Krawedz.hh.

4.8.2.2 Krawedz::Krawedz (string *psasiad*, int *pwaga* = 1) [inline]

Definition at line 20 of file Krawedz.hh.

4.8.3 Member Function Documentation

4.8.3.1 bool Krawedz::operator== (Krawedz *druga*)

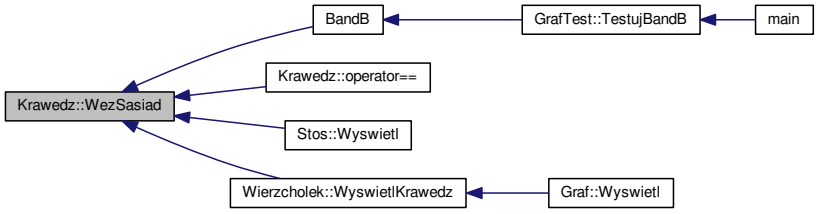
Definition at line 3 of file Krawedz.cpp.

Here is the call graph for this function:

4.8.3.2 string Krawedz::WezSasiad ( ) [inline]

Definition at line 21 of file Krawedz.hh.

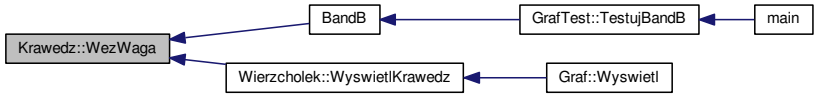
Here is the caller graph for this function:



4.8.3.3 int Krawedz::WezWaga ( ) [inline]

Definition at line 22 of file Krawedz.hh.

Here is the caller graph for this function:



4.8.4 Member Data Documentation

4.8.4.1 string Krawedz::sasiad ="0" [private]

Definition at line 14 of file Krawedz.hh.

4.8.4.2 int Krawedz::waga =1 [private]

Definition at line 15 of file Krawedz.hh.

The documentation for this class was generated from the following files:

- Krawedz.hh
- Krawedz.cpp

4.9 Lista< typ > Class Template Reference

`#include <Lista.hh>`

Public Member Functions

- bool Dodaj (typ elem, int ind)
- typ Usun (int ind)
- typ & Wez (int ind)

- int Rozmiar ()
- bool Czypusta ()
- void Oproznij ()
- void Wyswietl ()
- int Wyszukaj (typ szukane)

Private Attributes

- PojemnikWide< typ > * head =NULL
- PojemnikWide< typ > * tail =NULL

4.9.1 Detailed Description

template<typename typ>class Lista< typ >

Definition at line 18 of file Lista.hh.

4.9.2 Member Function Documentation

4.9.2.1 template<typename typ> bool Lista< typ >::Czypusta ( ) [inline]

Return values

true-	gdy lista jest pusta
false-	w przypadku przeciwnym

Definition at line 32 of file Lista.hh.

4.9.2.2 template<typename typ> bool Lista< typ >::Dodaj (typ *elem*, int *ind*)

Funkcja przypisuje wartosc do przechowania elementowi typu "Pojemnik" i dodaje ten "Pojeminik" w DOWOLNYM miejscu listy czyli na koncu, poczatku badz wewnatrz listy

Parameters

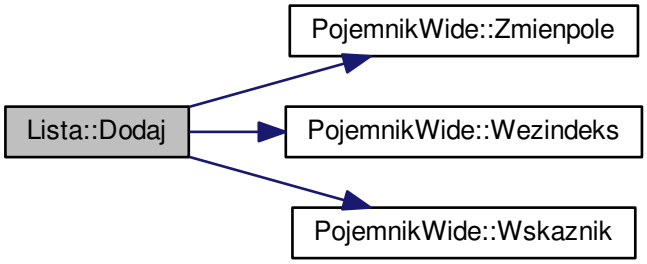
in	*elem*-	wartosc do przechowania
in	*index*-	indeks listy pod jakim bedzie przechowywany pojemnik ze zmienna

Return values

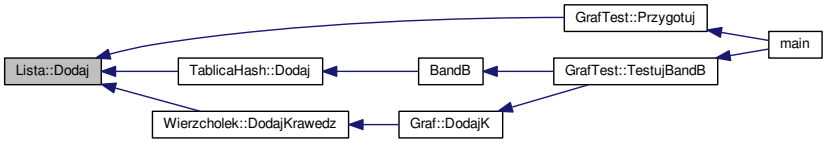
false-	gdy element ma byc wstawiony w nielogicznym miejscu, np-> wstawianie elementu o indeksie 100 kiedy lista ma aktualnie indeksy od 0 do 15
true-	gdy element wstawiono poprawnie do listy

Definition at line 53 of file Lista.hh.

Here is the call graph for this function:



Here is the caller graph for this function:

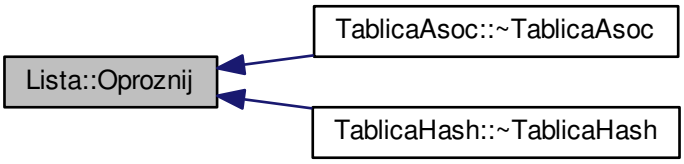


4.9.2.3 template<typename typ > void Lista< typ >::Oproznij ( )

Usuwa wszystkie elementy z listy

Definition at line 231 of file Lista.hh.

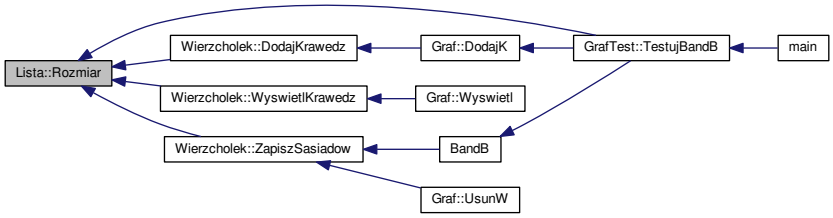
Here is the caller graph for this function:



4.9.2.4 template<typename typ> int Lista< typ >::Rozmiar ( ) [inline]

Definition at line 27 of file Lista.hh.

Here is the caller graph for this function:



4.9.2.5 template<typename typ > typ Lista< typ >::Usun (int *ind*)

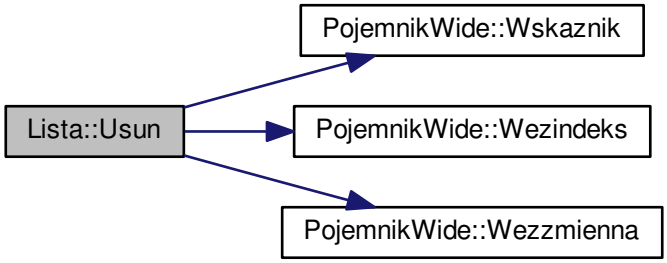
Usuwa element z Listy o zadanym indeksie i zwraca wartosc, ktora przechowywal

Parameters

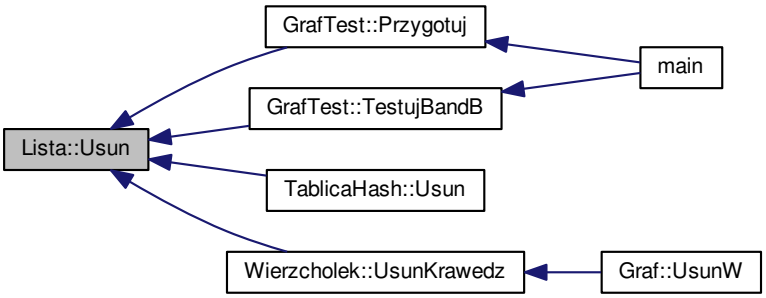
in	*ind-*	indeks elementu, ktory ma zostac usuniety z listy

Definition at line 146 of file Lista.hh.

Here is the call graph for this function:



Here is the caller graph for this function:



4.9.2.6 template<typename typ > typ & Lista< typ >::Wez (int *ind*)

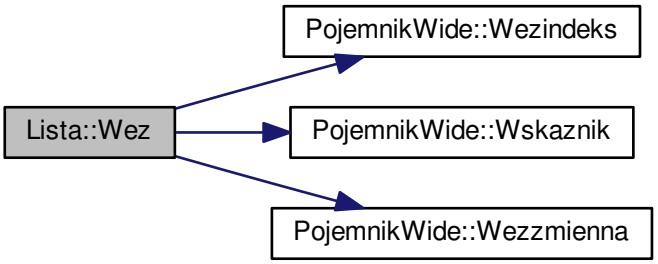
Zwraca wartość elementu o zadanym indeksie

Parameters

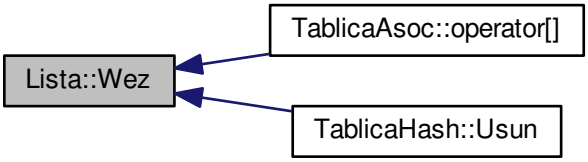
in	*ind*-	indeks poszukiwanego elementu

Definition at line 118 of file Lista.hh.

Here is the call graph for this function:



Here is the caller graph for this function:

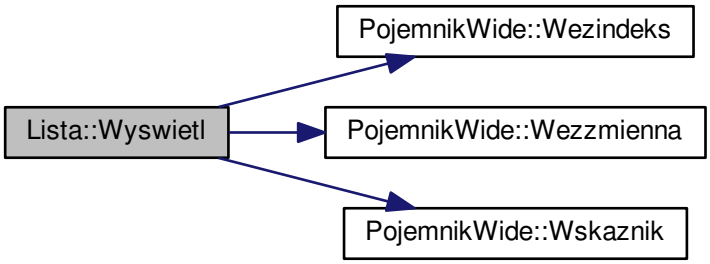


4.9.2.7 template<typename typ > void Lista< typ >::Wyswietl ( )

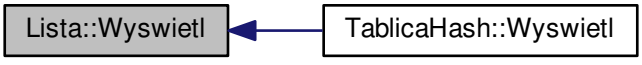
Wyswietla zawartosc listy na standardowe wyjscie

Definition at line 247 of file Lista.hh.

Here is the call graph for this function:



Here is the caller graph for this function:



4.9.2.8 template<typename typ> int Lista< typ >::Wyszukaj (typ *szukane*)

Wyszukuje podany wyraz wsrod elementow listy

Parameters

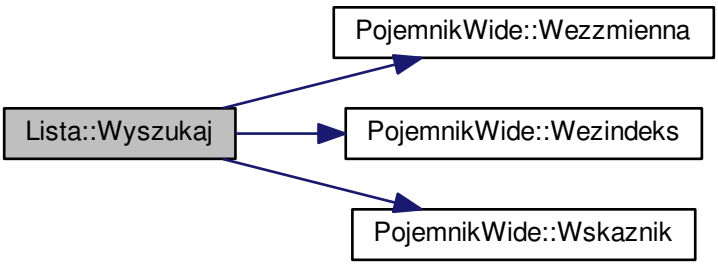
in	*szukane-*	szukany wyraz

Return values

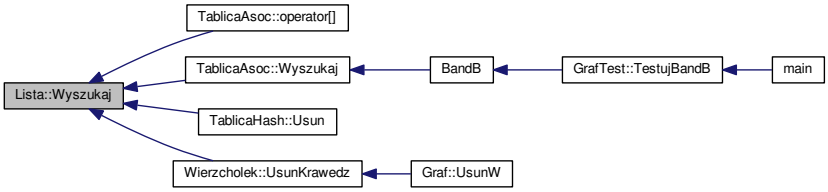
zwraca	numer indeksu elementu, ktory przechowuje szukany wyraz lub -1 w przypadku jego nieznalezienia

Definition at line 274 of file Lista.hh.

Here is the call graph for this function:



Here is the caller graph for this function:



4.9.3 Member Data Documentation

4.9.3.1 template<typename typ> PojemnikWide<typ>* Lista< typ >::head =NULL [private]

Definition at line 19 of file Lista.hh.

4.9.3.2 template<typename typ> PojemnikWide<typ>* Lista< typ >::tail =NULL [private]

Definition at line 20 of file Lista.hh.

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

- Lista.hh

4.10 Pojemnik Class Reference

`#include <Pojemnik.hh>`

Collaboration diagram for Pojemnik:

Public Member Functions

- void Zmienpole (string pom)
- string Wezzmienna ()
- Pojemnik ∗ Wskaznik ()

Public Attributes

- Pojemnik ∗ wsk =NULL

Private Attributes

- string zmienna

4.10.1 Detailed Description

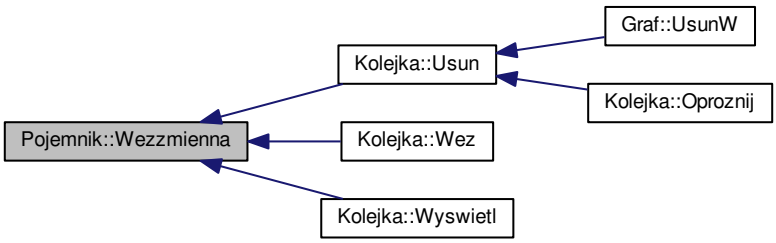
Definition at line 13 of file Pojemnik.hh.

4.10.2 Member Function Documentation

4.10.2.1 string Pojemnik::Wezzmienna ( ) [inline]

Definition at line 19 of file Pojemnik.hh.

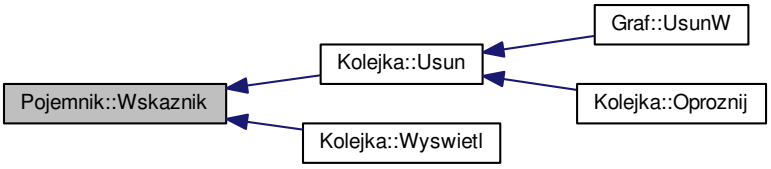
Here is the caller graph for this function:



4.10.2.2 Pojemnik* Pojemnik::Wskaznik ( ) [inline]

Definition at line 20 of file Pojemnik.hh.

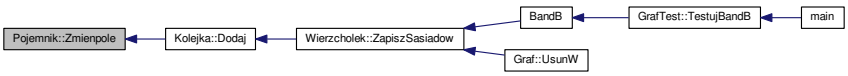
Here is the caller graph for this function:



4.10.2.3 void Pojemnik::Zmienpole (string *pom*) [inline]

Definition at line 18 of file Pojemnik.hh.

Here is the caller graph for this function:



4.10.3 Member Data Documentation

4.10.3.1 Pojemnik* Pojemnik::wsk =NULL

Definition at line 16 of file Pojemnik.hh.

4.10.3.2 string Pojemnik::zmienna [private]

Definition at line 14 of file Pojemnik.hh.

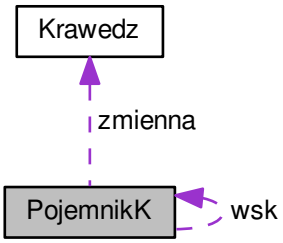
The documentation for this class was generated from the following file:

- Pojemnik.hh

4.11 PojemnikK Class Reference

`#include <PojemnikK.hh>`

Collaboration diagram for PojemnikK:



Public Member Functions

- void Zmienpole (Krawedz pom)
- Krawedz Wezzmienna ()
- PojemnikK * Wskaznik ()

Public Attributes

- PojemnikK * wsk =NULL

Private Attributes

- Krawedz zmienna

4.11.1 Detailed Description

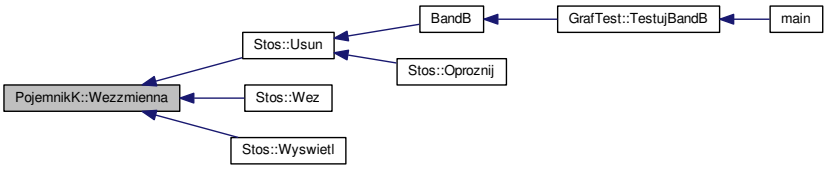
Definition at line 14 of file PojemnikK.hh.

4.11.2 Member Function Documentation

4.11.2.1 Krawedz PojemnikK::Wezzmienna ( ) [inline]

Definition at line 20 of file PojemnikK.hh.

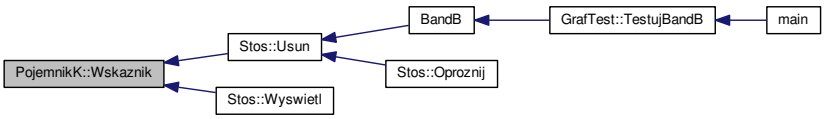
Here is the caller graph for this function:



4.11.2.2 PojemnikK* PojemnikK::Wskaznik ( ) [inline]

Definition at line 21 of file PojemnikK.hh.

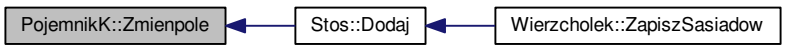
Here is the caller graph for this function:



4.11.2.3 void PojemnikK::Zmienpole (Krawedz *pom*) [inline]

Definition at line 19 of file PojemnikK.hh.

Here is the caller graph for this function:



4.11.3 Member Data Documentation

4.11.3.1 PojemnikK* PojemnikK::wsk =NULL

Definition at line 17 of file PojemnikK.hh.

4.11.3.2 Krawedz PojemnikK::zmienna [private]

Definition at line 15 of file PojemnikK.hh.

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

- PojemnikK.hh

4.12 PojemnikWide< typ > Class Template Reference

`#include <PojemnikWide.hh>`

Public Member Functions

- void Zmienpole (typ pom)
- int & Wezindeks ()
- typ & Wezzmienna ()
- PojemnikWide< typ > * Wskaznik ()

Public Attributes

- PojemnikWide< typ > * wsk =NULL

Private Attributes

- typ zmienna
- int indeks =0

4.12.1 Detailed Description

template<typename typ>class PojemnikWide< typ >

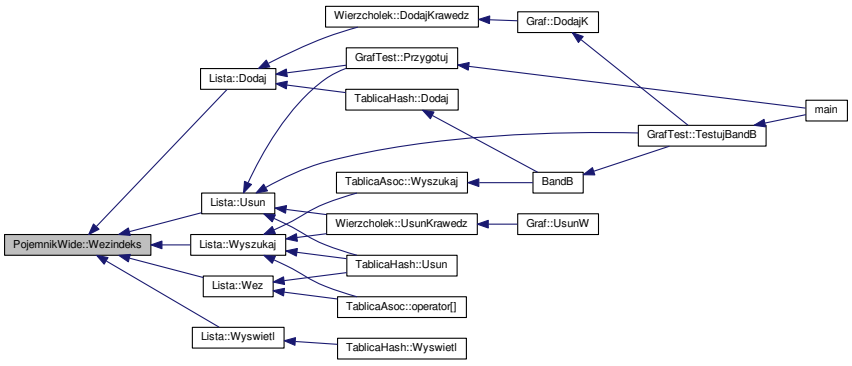
Definition at line 13 of file PojemnikWide.hh.

4.12.2 Member Function Documentation

4.12.2.1 template<typename typ> int& PojemnikWide< typ >::Wezindeks ( ) [inline]

Definition at line 20 of file PojemnikWide.hh.

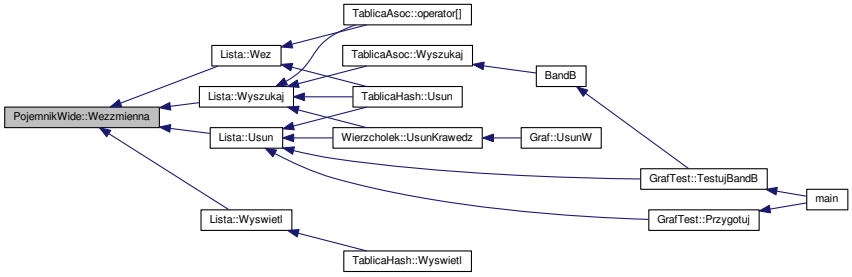
Here is the caller graph for this function:



4.12.2.2 template<typename typ> typ& PojemnikWide< typ >::Wezzmienna ( ) [inline]

Definition at line 21 of file PojemnikWide.hh.

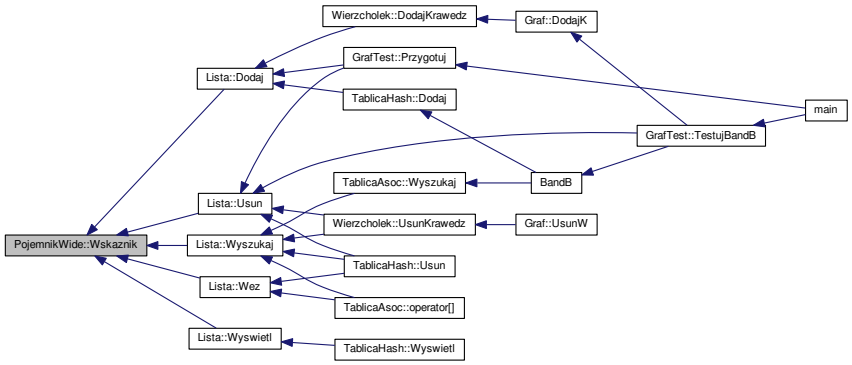
Here is the caller graph for this function:



4.12.2.3 template<typename typ> PojemnikWide<typ>* PojemnikWide< typ >::Wskaznik ( ) [inline]

Definition at line 22 of file PojemnikWide.hh.

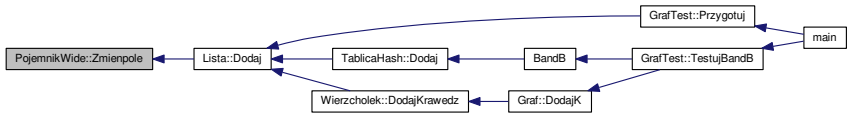
Here is the caller graph for this function:



4.12.2.4 template<typename typ> void PojemnikWide< typ >::Zmienpole (typ *pom*) [inline]

Definition at line 19 of file PojemnikWide.hh.

Here is the caller graph for this function:



4.12.3 Member Data Documentation

4.12.3.1 template<typename typ> int PojemnikWide< typ >::indeks =0 [private]

Definition at line 15 of file PojemnikWide.hh.

4.12.3.2 template<typename typ> PojemnikWide<typ>* PojemnikWide< typ >::wsk =NULL

Definition at line 17 of file PojemnikWide.hh.

4.12.3.3 template<typename typ> typ PojemnikWide< typ >::zmienna [private]

Definition at line 14 of file PojemnikWide.hh.

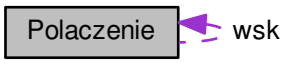
The documentation for this class was generated from the following file:

- PojemnikWide.hh

4.13 Polaczenie Class Reference

`#include <Polaczenie.hh>`

Collaboration diagram for Polaczenie:



Public Member Functions

- string & WezPozycja ()
- int & WezSuma ()
- string & WezHistoria ()
- Polaczenie * Wskaznik ()

Public Attributes

- Polaczenie * wsk =NULL

Private Attributes

- string pozycja ="brak"
- int suma =0
- string historia =""

4.13.1 Detailed Description

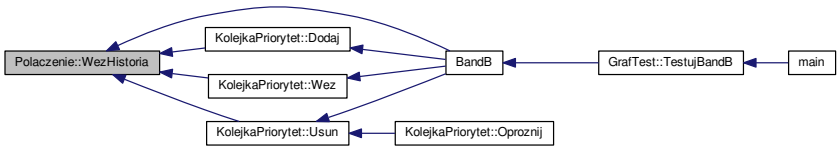
Definition at line 13 of file Polaczenie.hh.

4.13.2 Member Function Documentation

4.13.2.1 string& Polaczenie::WezHistoria ( ) [inline]

Definition at line 22 of file Polaczenie.hh.

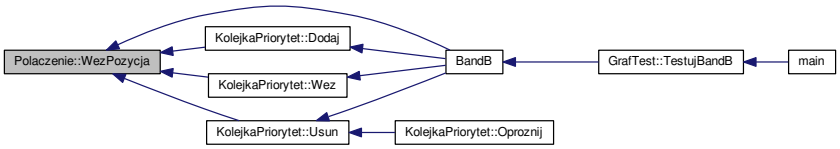
Here is the caller graph for this function:



4.13.2.2 string& Polaczenie::WezPozycja ( ) [inline]

Definition at line 20 of file Polaczenie.hh.

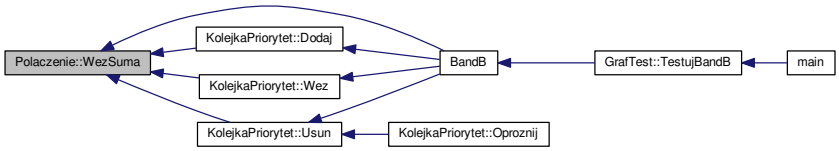
Here is the caller graph for this function:



4.13.2.3 int& Polaczenie::WezSuma ( ) [inline]

Definition at line 21 of file Polaczenie.hh.

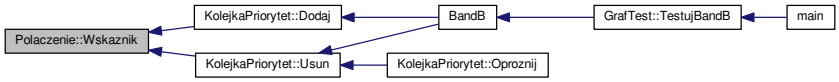
Here is the caller graph for this function:



4.13.2.4 Polaczenie* Polaczenie::Wskaznik ( ) [inline]

Definition at line 23 of file Polaczenie.hh.

Here is the caller graph for this function:



4.13.3 Member Data Documentation

4.13.3.1 string Polaczenie::historia ="" [private]

Definition at line 16 of file Polaczenie.hh.

4.13.3.2 string Polaczenie::pozycja ="brak" [private]

Definition at line 14 of file Polaczenie.hh.

4.13.3.3 int Polaczenie::suma =0 [private]

Definition at line 15 of file Polaczenie.hh.

4.13.3.4 Polaczenie* Polaczenie::wsk =NULL

Definition at line 18 of file Polaczenie.hh.

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

- Polaczenie.hh

4.14 Rekord Class Reference

`#include <Rekord.hh>`

Public Member Functions

- Rekord ()
- Rekord (string pklucz)
- Rekord (string pklucz, int pnumer)
- string & WezKlucz ()
- int & WezNumer ()

Private Attributes

- string klucz
- int numer =0

4.14.1 Detailed Description

Definition at line 13 of file Rekord.hh.

4.14.2 Constructor & Destructor Documentation

4.14.2.1 Rekord::Rekord ( ) [inline]

Definition at line 17 of file Rekord.hh.

4.14.2.2 Rekord::Rekord (string *pklucz*) [inline]

Definition at line 18 of file Rekord.hh.

4.14.2.3 Rekord::Rekord (string *pklucz*, int *pnumer*) [inline]

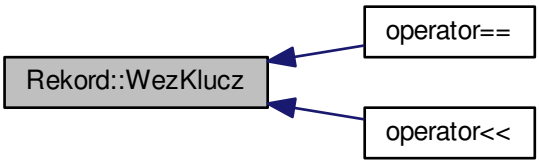
Definition at line 19 of file Rekord.hh.

4.14.3 Member Function Documentation

4.14.3.1 string& Rekord::WezKlucz ( ) [inline]

Definition at line 20 of file Rekord.hh.

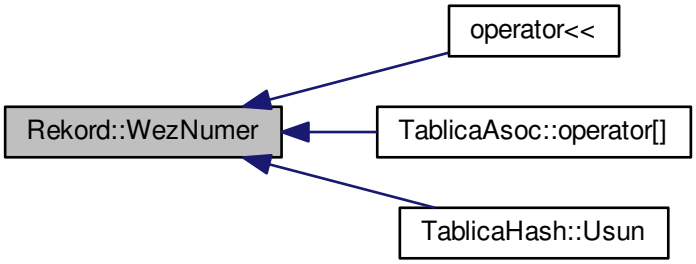
Here is the caller graph for this function:



4.14.3.2 int& Rekord::WezNumer ( ) [inline]

Definition at line 21 of file Rekord.hh.

Here is the caller graph for this function:



4.14.4 Member Data Documentation

4.14.4.1 string Rekord::klucz [private]

Definition at line 14 of file Rekord.hh.

4.14.4.2 int Rekord::numer =0 [private]

Definition at line 15 of file Rekord.hh.

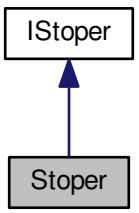
The documentation for this class was generated from the following file:

- Rekord.hh

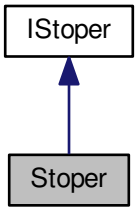
4.15 Stoper Class Reference

`#include <Stoper.hh>`

Inheritance diagram for Stoper:



Collaboration diagram for Stoper:



Public Member Functions

- void start ()
- void stop ()
- double get_time ()

  roznica czasowa
- bool zapis_do_pliku ()

  Zapis zmierzonego czasu do pliku.

Private Attributes

- timeval czas1
- timeval czas2

4.15.1 Detailed Description

Definition at line 12 of file Stoper.hh.

4.15.2 Member Function Documentation

4.15.2.1 double Stoper::get_time ( ) [virtual]

Zwraca roznice czasu miedzy "startem a "stopem". Wartosci wyrazone w mikrosekundach

Implements IStoper.

Definition at line 9 of file Stoper.cpp.

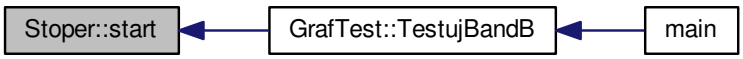
Here is the caller graph for this function:

4.15.2.2 void Stoper::start ( ) [inline], [virtual]

Implements IStoper.

Definition at line 17 of file Stoper.hh.

Here is the caller graph for this function:

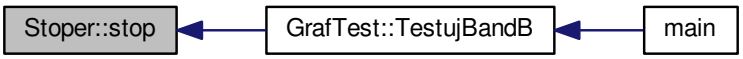


4.15.2.3 void Stoper::stop ( ) [inline], [virtual]

Implements IStoper.

Definition at line 18 of file Stoper.hh.

Here is the caller graph for this function:



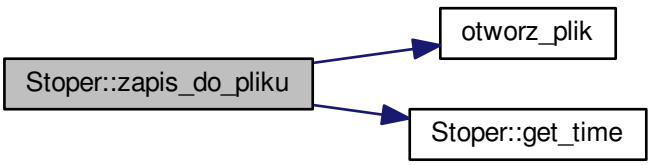
4.15.2.4 bool Stoper::zapis_do_pliku ( ) [virtual]

Wywolanie tej funkcji skutkuje dopisaniem do pliku "czasy.dat" ostatniej roznicy czasowej ("czas_stop"-"czas_start) Wartosci wyrazone w sekundach

Implements IStoper.

Definition at line 43 of file Stoper.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

4.15.3 Member Data Documentation

4.15.3.1 timeval Stoper::czas1 [private]

Definition at line 13 of file Stoper.hh.

4.15.3.2 timeval Stoper::czas2 [private]

Definition at line 14 of file Stoper.hh.

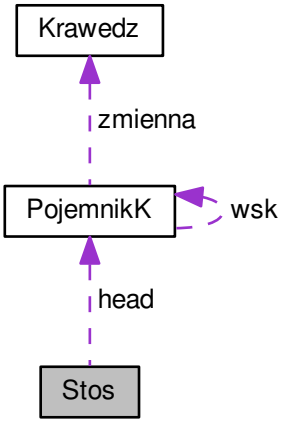
The documentation for this class was generated from the following files:

- Stoper.hh
- Stoper.cpp

4.16 Stos Class Reference

`#include <Stos.hh>`

Collaboration diagram for Stos:



Public Member Functions

- ~Stos ()
- void Dodaj (Krawedz elem)
- Krawedz Usun ()
- Krawedz Wez ()
- bool Czypusty ()
- int Rozmiar ()
- void Oproznij ()
- void Wyswietl ()

Private Attributes

- PojemnikK * head =NULL
- int rozmiar =0

4.16.1 Detailed Description

Definition at line 13 of file Stos.hh.

4.16.2 Constructor & Destructor Documentation

4.16.2.1 Stos::~Stos ( ) [inline]

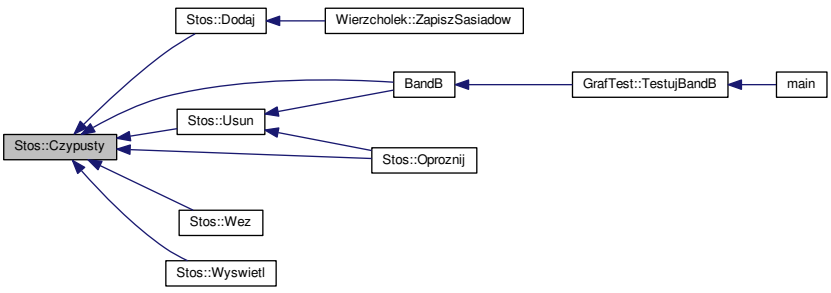
Definition at line 17 of file Stos.hh.

4.16.3 Member Function Documentation

4.16.3.1 bool Stos::Czypusty ( ) [inline]

Definition at line 21 of file Stos.hh.

Here is the caller graph for this function:



4.16.3.2 void Stos::Dodaj (Krawedz *elem*)

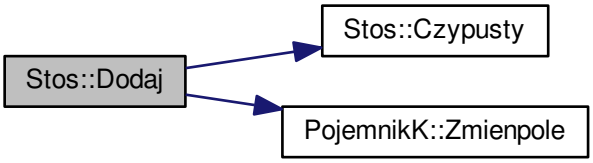
"Kladzie" element na Stos

Parameters

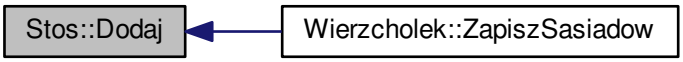
in	*elem-*	zmienna do przechowania

Definition at line 13 of file Stos.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

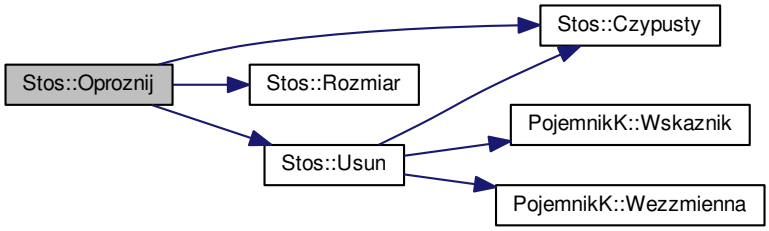


4.16.3.3 void Stos::Oproznij ( )

Usuwa wszystkie elementy stosu

Definition at line 71 of file Stos.cpp.

Here is the call graph for this function:



4.16.3.4 int Stos::Rozmiar ( ) [inline]

Definition at line 22 of file Stos.hh.

Here is the caller graph for this function:

4.16.3.5 Krawedz Stos::Usun ( )

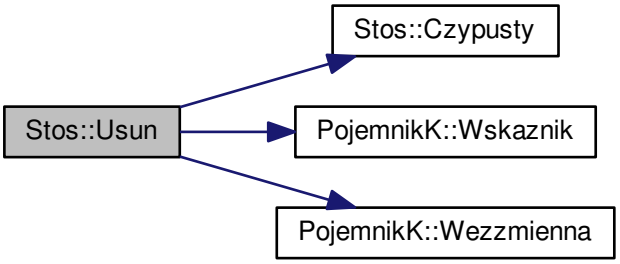
Usuwa element ze stosu

Return values

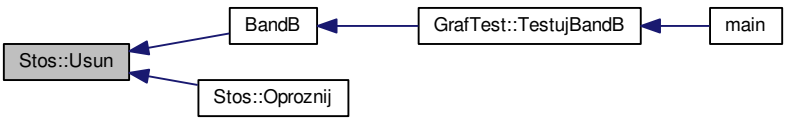
wartosc	usunietego elementu

Definition at line 35 of file Stos.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

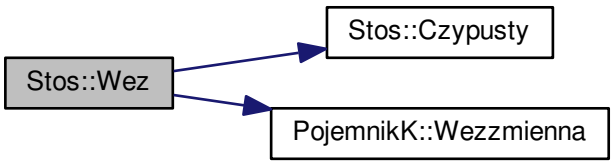


4.16.3.6 Krawedz Stos::Wez ( )

Zwraca wartosc elementu stosu, ktory jest "na wierzchu". Funkcja NIE sluzy do modyfikowania wartosci tego elementu

Definition at line 57 of file Stos.cpp.

Here is the call graph for this function:

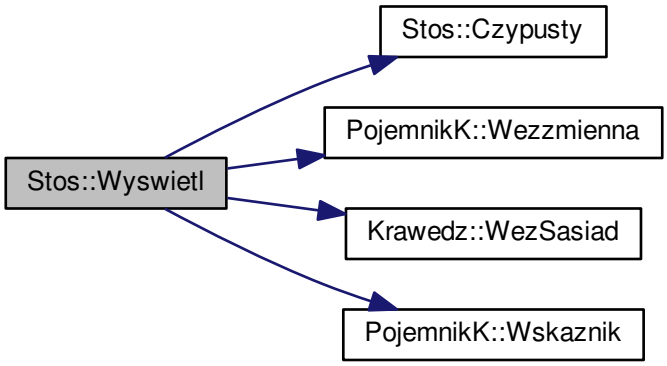


4.16.3.7 void Stos::Wyswietl ( )

Wyswietla wszystkie elementy stosu od "wierzcholka" do dolu

Definition at line 82 of file Stos.cpp.

Here is the call graph for this function:



4.16.4 Member Data Documentation

4.16.4.1 PojemnikK* Stos::head =NULL [private]

Definition at line 14 of file Stos.hh.

4.16.4.2 int Stos::rozmiar =0 [private]

Definition at line 15 of file Stos.hh.

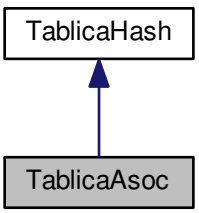
The documentation for this class was generated from the following files:

- Stos.hh
- Stos.cpp

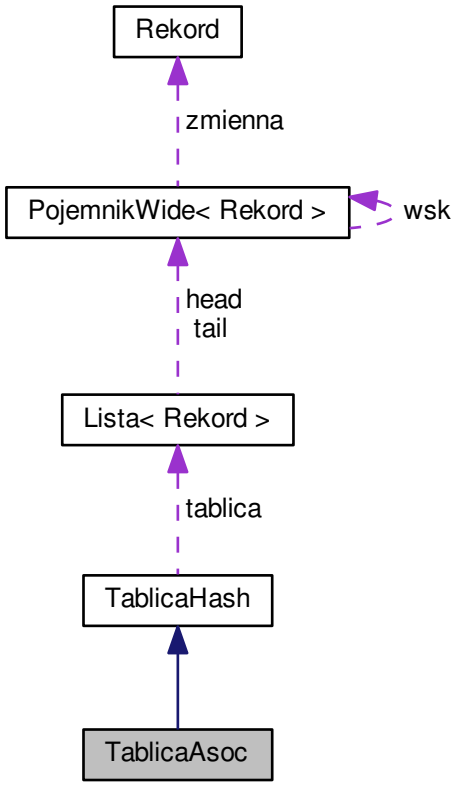
4.17 TablicaAsoc Class Reference

`#include <TablicaAsoc.hh>`

Inheritance diagram for TablicaAsoc:



Collaboration diagram for TablicaAsoc:



Public Member Functions

- TablicaAsoc ()
- TablicaAsoc (int prozmiar)
- ~TablicaAsoc ()

  Destruktor.
- int operator[] (string szukanyklucz)
- bool Wyszukaj (string szukanyklucz)

4.17.1 Detailed Description

Definition at line 11 of file TablicaAsoc.hh.

4.17.2 Constructor & Destructor Documentation

4.17.2.1 TablicaAsoc::TablicaAsoc ( ) [inline]

Definition at line 14 of file TablicaAsoc.hh.

4.17.2.2 TablicaAsoc::TablicaAsoc (int *prozmiar*) [inline]

Definition at line 15 of file TablicaAsoc.hh.

4.17.2.3 TablicaAsoc::~TablicaAsoc ( )

Usuwa wszystkie elementy tablicy

Definition at line 14 of file TablicaAsoc.cpp.

Here is the call graph for this function:

4.17.3 Member Function Documentation

4.17.3.1 int TablicaAsoc::operator[] (string *szukanyklucz*)

Metoda wyszukujaca numer dla podanego klucza

Parameters

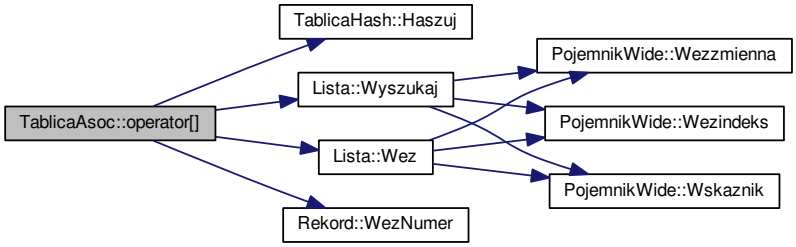
in	*szukanyklucz-*	dla tego klucza szukamy numer

Return values

numer	telefonu dla szukanego klucza

Definition at line 30 of file TablicaAsoc.cpp.

Here is the call graph for this function:



4.17.3.2 bool TablicaAsoc::Wyszukaj (string *szukanyklucz*)

Metoda sprwadzajaca czy tablica przechowuje rekord o podanym kluczu

Parameters

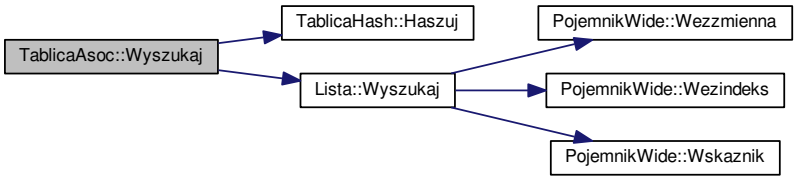
in	*szukanyklucz-*	rekord z takim kluczem jest szukany

Return values

true	jezeli tablica przechowuje rekord o podanym kluczu

Definition at line 48 of file TablicaAsoc.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

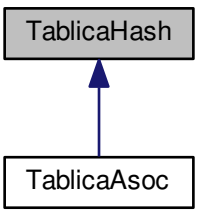
The documentation for this class was generated from the following files:

- TablicaAsoc.hh
- TablicaAsoc.cpp

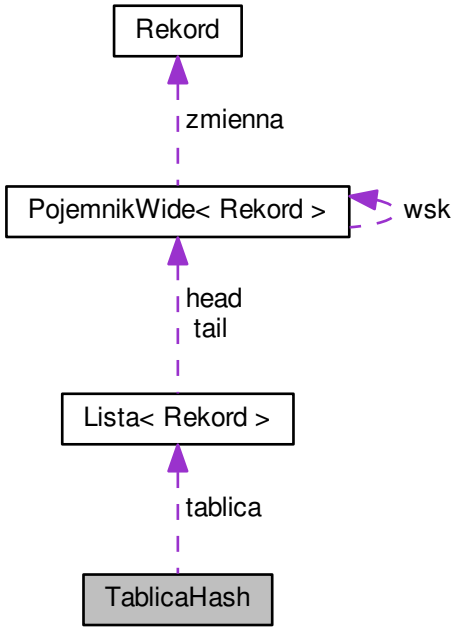
4.18 TablicaHash Class Reference

`#include <TablicaHash.hh>`

Inheritance diagram for TablicaHash:



Collaboration diagram for TablicaHash:



Public Member Functions

- TablicaHash ()
- TablicaHash (int prozmiar)
- ~TablicaHash ()

  Destruktor.
- int Haszuj (string nazwa)
- bool Dodaj (string klucz, int numer=1)
- int Usun (string klucz)
- void Wyswietl ()

Private Attributes

- Lista< Rekord > * tablica
- int rozmiar =10

Friends

- class TablicaAsoc

4.18.1 Detailed Description

Definition at line 14 of file TablicaHash.hh.

4.18.2 Constructor & Destructor Documentation

4.18.2.1 TablicaHash::TablicaHash ( ) [inline]

Definition at line 21 of file TablicaHash.hh.

4.18.2.2 TablicaHash::TablicaHash (int *prozmiar*) [inline]

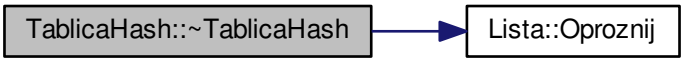
Definition at line 22 of file TablicaHash.hh.

4.18.2.3 TablicaHash::~TablicaHash ( )

Usuwa wszystkie elementy tablicy

Definition at line 13 of file TablicaHash.cpp.

Here is the call graph for this function:



4.18.3 Member Function Documentation

4.18.3.1 bool TablicaHash::Dodaj (string *klucz*, int *numer* = 1)

Funkcja dodajaca zestaw danych do tablicy haszujacej

Parameters

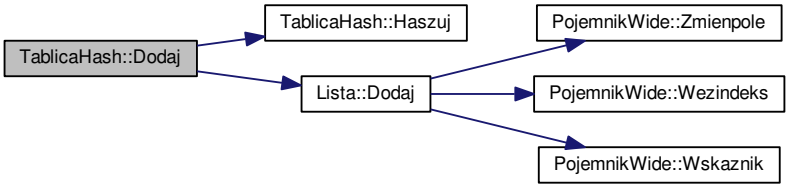
in	*klucz-na*	jego podstawie funkcja dobiera indeks tablicy gdzie maja byc zapisane dane
in	*numer-*	jedna z dwoch danych do przechowania

Return values

true-jesli	dodawanie do tablicy powiodlo sie

Definition at line 46 of file TablicaHash.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

4.18.3.2 int TablicaHash::Haszuj (string *nazwa*)

Mapuje klucz typu string na integer w stalym przedziale [0,rozmiartablicy-1]

Parameters

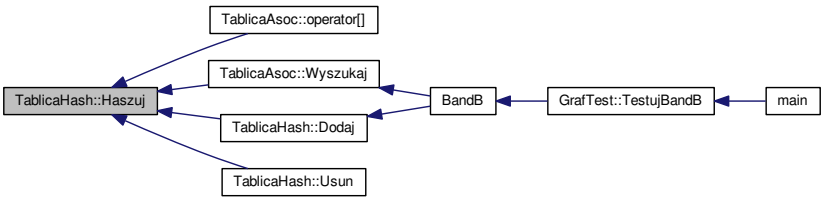
in	*nazwa-klucz*	do mapowania

Return values

indeks	tablicy, do ktorej maja byc zapisane dane

Definition at line 29 of file TablicaHash.cpp.

Here is the caller graph for this function:



4.18.3.3 int TablicaHash::Usun (string *klucz*)

Usuwa element z tablicy Haszujacej o danym kluczu i zwraca numer powiazany z tym kluczem

Parameters

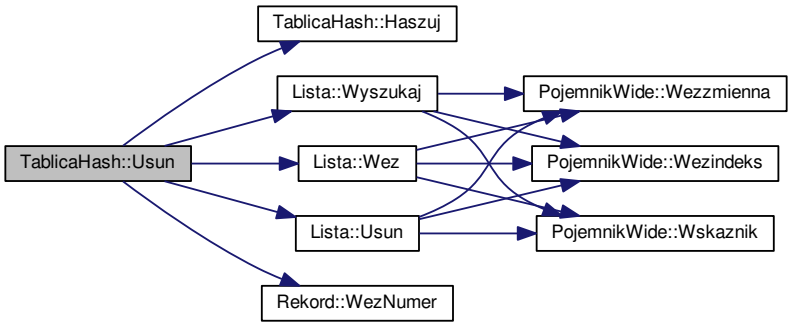
in	*klucz-*	element o takim kluczu zostanie usuniety

Return values

Numer	powiazany z kluczem, ktory zostanie razem z nim usuniety lub -1 gdy element o podanym kluczu nie istnieje

Definition at line 60 of file TablicaHash.cpp.

Here is the call graph for this function:

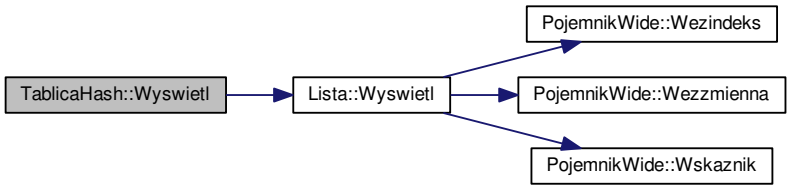


4.18.3.4 void TablicaHash::Wyswietl ( )

Wyswietla na standardowe wyjscie zwartosc tablicy haszujacej

Definition at line 76 of file TablicaHash.cpp.

Here is the call graph for this function:



4.18.4 Friends And Related Function Documentation

4.18.4.1 friend class TablicaAsoc [friend]

Definition at line 15 of file TablicaHash.hh.

4.18.5 Member Data Documentation

4.18.5.1 int TablicaHash::rozmiar =10 [private]

Definition at line 18 of file TablicaHash.hh.

4.18.5.2 Lista<Rekord>* TablicaHash::tablica [private]

Definition at line 17 of file TablicaHash.hh.

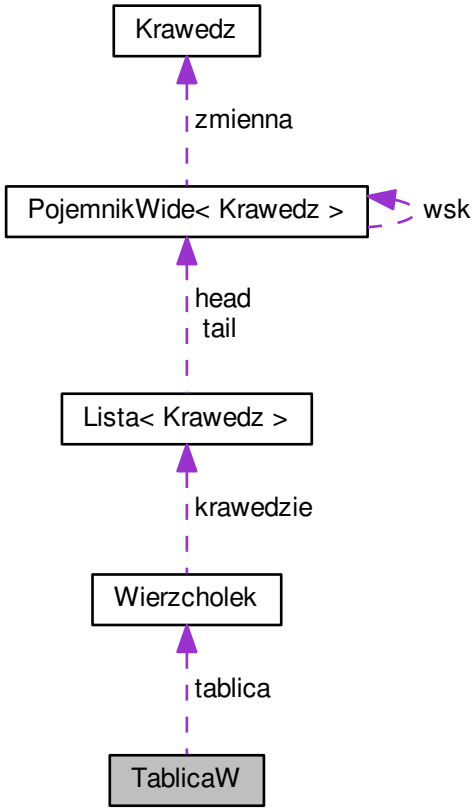
The documentation for this class was generated from the following files:

- TablicaHash.hh
- TablicaHash.cpp

4.19 TablicaW Class Reference

`#include <TablicaW.hh>`

Collaboration diagram for TablicaW:



Public Member Functions

- TablicaW ()
- ~TablicaW ()

  Destruktor.
- int WezRozmiar ()
- int WezPelne ()
- void DodajW (Wierzcholek W)
- int Wyszukaj (string szukane)
- Wierzcholek & WezW (int indeks)

Private Attributes

- Wierzcholek * tablica =NULL
- int rozmiar =10
- int pelne =0

4.19.1 Detailed Description

Definition at line 13 of file TablicaW.hh.

4.19.2 Constructor & Destructor Documentation

4.19.2.1 TablicaW::TablicaW ( ) [inline]

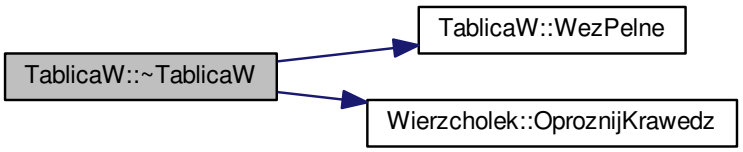
Definition at line 19 of file TablicaW.hh.

4.19.2.2 TablicaW::~TablicaW ( )

Usuwa wszystkie elementy z tablicy

Definition at line 8 of file TablicaW.cpp.

Here is the call graph for this function:



4.19.3 Member Function Documentation

4.19.3.1 void TablicaW::DodajW (Wierzcholek *W*)

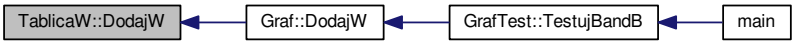
Dodaje wierzcholek do tablicy wierzcholkow

Parameters

in	*W*	Wierzcholek do zapisania w tablicy wierzcholkow

Definition at line 21 of file TablicaW.cpp.

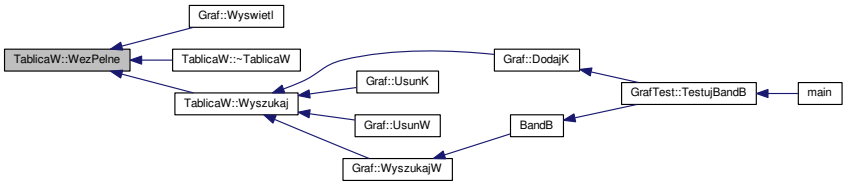
Here is the caller graph for this function:



4.19.3.2 int TablicaW::WezPelne ( ) [inline]

Definition at line 22 of file TablicaW.hh.

Here is the caller graph for this function:



4.19.3.3 int TablicaW::WezRozmiar ( ) [inline]

Definition at line 21 of file TablicaW.hh.

4.19.3.4 Wierzcholek& TablicaW::WezW (int *indeks*) [inline]

Definition at line 26 of file TablicaW.hh.

4.19.3.5 int TablicaW::Wyszukaj (string *szukane*)

Wyszukiwanie podanego wierzcholka w tablicy wierzcholkow

Parameters

in	*szukane*	nazwa szukanego wierzcholka

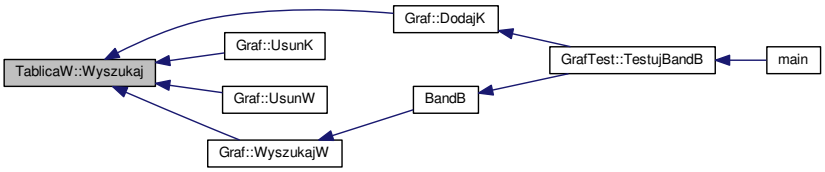
Return values

indeks	komorki tablicy, ktora przechowuje szukany wierzcholek lub -1 jezeli nie znaleziono danego wierzchoka

Definition at line 55 of file TablicaW.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:



4.19.4 Member Data Documentation

4.19.4.1 int TablicaW::pelne =0 [private]

Definition at line 16 of file TablicaW.hh.

4.19.4.2 int TablicaW::rozmiar =10 [private]

Definition at line 15 of file TablicaW.hh.

4.19.4.3 Wierzcholek* TablicaW::tablica =NULL [private]

Definition at line 14 of file TablicaW.hh.

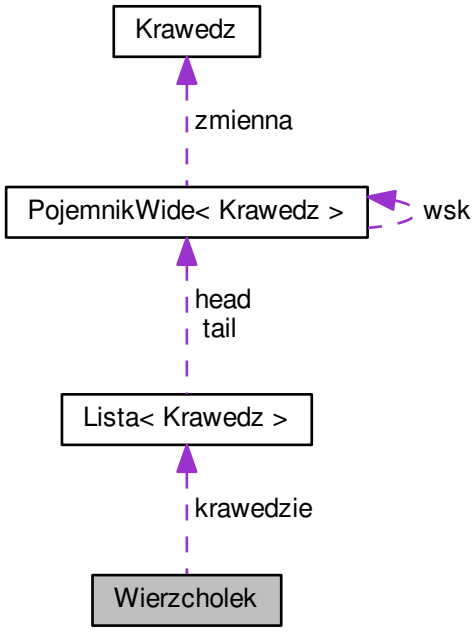
The documentation for this class was generated from the following files:

- TablicaW.hh
- TablicaW.cpp

4.20 Wierzcholek Class Reference

`#include <Wierzcholek.hh>`

Collaboration diagram for Wierzcholek:



Public Member Functions

- Wierzcholek ()
- Wierzcholek (string pnazwa)
- string & Pozycja ()
- void DodajKrawedz (string sasiad, int waga=1)
- bool UsunKrawedz (string sasiad)
- Krawedz & WezK (int indeks)
- void WyswietlKrawedz ()
- void OproznijKrawedz ()
- void ZapiszSasiadow (Kolejka &kolejka)
- void ZapiszSasiadow (Stos &stos)
- bool operator== (Wierzcholek drugi)

Private Attributes

- string nazwa ="1"
- Lista< Krawedz > krawedzie

4.20.1 Detailed Description

Definition at line 11 of file Wierzcholek.hh.

4.20.2 Constructor & Destructor Documentation

4.20.2.1 Wierzcholek::Wierzcholek ( ) [inline]

Definition at line 16 of file Wierzcholek.hh.

4.20.2.2 Wierzcholek::Wierzcholek (string *pnazwa*) [inline]

Definition at line 17 of file Wierzcholek.hh.

4.20.3 Member Function Documentation

4.20.3.1 void Wierzcholek::DodajKrawedz (string *sasiad,* int *waga* = 1)

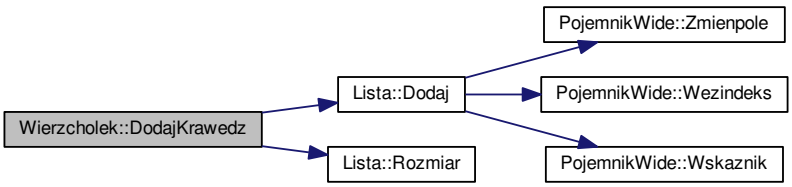
Przypisuje krawedz o zadanych parametrach do wierzcholka

Parameters

in	*sasiad*	zostanie dodana krawedz prowadzaca do tego sasiada
in	*waga*	waga krawedzi

Definition at line 9 of file Wierzcholek.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

4.20.3.2 bool Wierzcholek::operator== (Wierzcholek *drugi*)

Przeciazenie porownania na potrzeby wyszukiwania wierzcholka

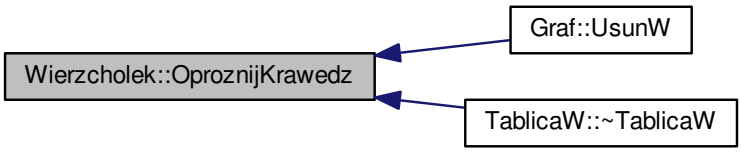
Definition at line 47 of file Wierzcholek.cpp.

Here is the call graph for this function:

4.20.3.3 void Wierzcholek::OproznijKrawedz ( ) [inline]

Definition at line 27 of file Wierzcholek.hh.

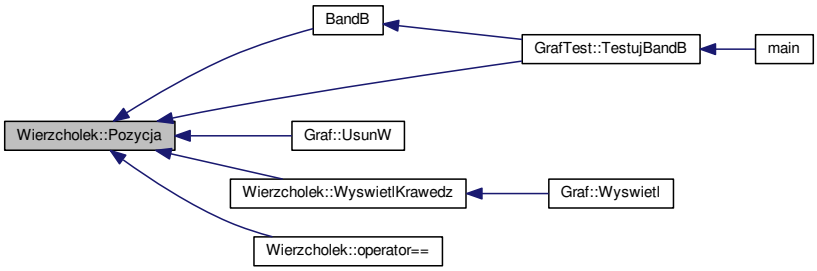
Here is the caller graph for this function:



4.20.3.4 string& Wierzcholek::Pozycja ( ) [inline]

Definition at line 19 of file Wierzcholek.hh.

Here is the caller graph for this function:



4.20.3.5 bool Wierzcholek::UsunKrawedz (string *sasiad*)

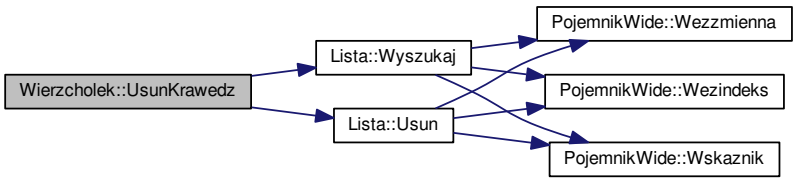
Usuwa podana krawedz przylegajaca do wierzcholka

Parameters

in	*sasiad*	krawedz prowadzaca do tego sasiada mamy usunac

Definition at line 19 of file Wierzcholek.cpp.

Here is the call graph for this function:



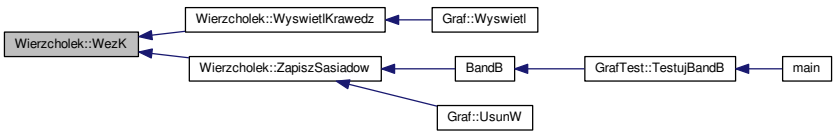
Here is the caller graph for this function:

4.20.3.6 Krawedz& Wierzcholek::WezK (int *indeks*) [inline]

Daje dostep do i'tej krawedzi zapisanej na liscie wierzcholka

Definition at line 25 of file Wierzcholek.hh.

Here is the caller graph for this function:

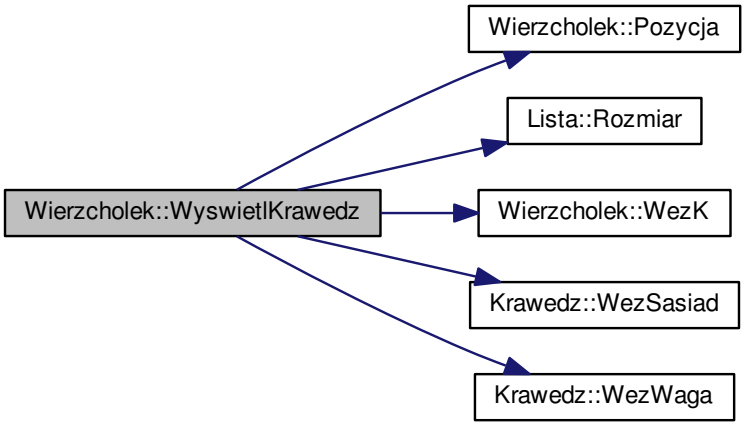


4.20.3.7 void Wierzcholek::WyswietlKrawedz ( )

Wyswietla wszystkie krawedzie przylegajace do danego wierzcholka

Definition at line 34 of file Wierzcholek.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

4.20.3.8 void Wierzcholek::ZapiszSasiadow (Kolejka & *kolejka*)

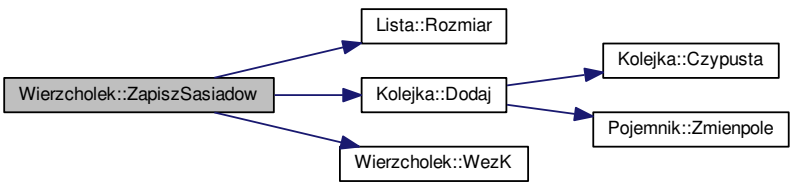
Zapisuje w kolejce wszystkie sasiadujace wierzcholki

Parameters

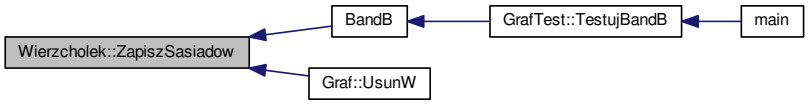
in	*kolejka*	w niej zapisane beda informacje o sasiadach

Definition at line 60 of file Wierzcholek.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



4.20.3.9 void Wierzcholek::ZapiszSasiadow (Stos & *stos*)

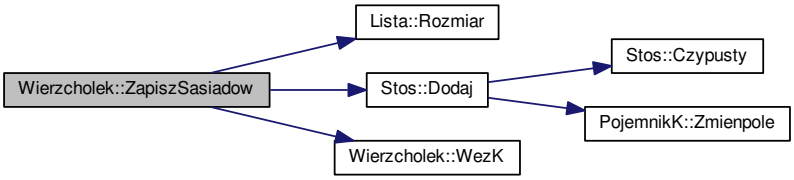
Zapisuje do stosu krawedzie przylegajace do wierzcholka. Przechowuja one informacje o sasiadach wierzcholka oraz o koszcie przejscia do danego sasiada

Parameters

in	*stos*	w nim zapisane beda krawedzie

Definition at line 72 of file Wierzcholek.cpp.

Here is the call graph for this function:



4.20.4 Member Data Documentation

4.20.4.1 Lista<Krawedz> Wierzcholek::krawedzie [private]

Definition at line 13 of file Wierzcholek.hh.

4.20.4.2 string Wierzcholek::nazwa ="1" [private]

Definition at line 12 of file Wierzcholek.hh.

The documentation for this class was generated from the following files:

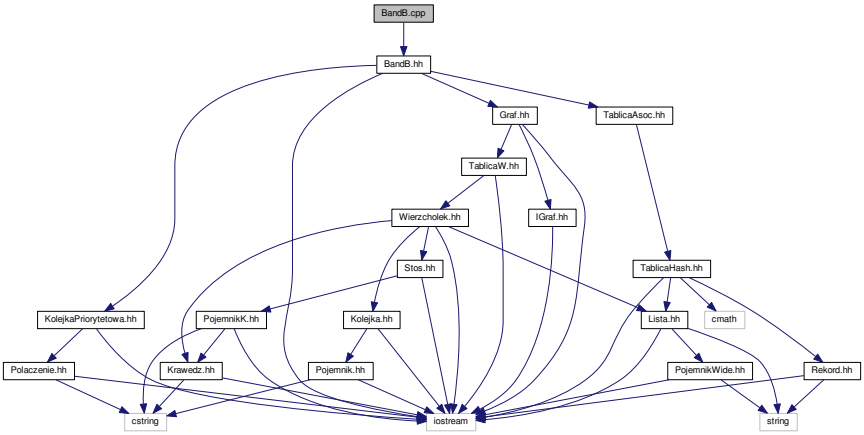
- Wierzcholek.hh
- Wierzcholek.cpp

Chapter 5

File Documentation

5.1 BandB.cpp File Reference

`#include "BandB.hh"`
Include dependency graph for BandB.cpp:



Functions

- int BandB (Graf &graf, string start, string meta)

  Branch and Bound.

5.1.1 Function Documentation

5.1.1.1 int BandB (Graf & *graf,* string *start,* string *meta*)

Algorytm znajduje najkrotsza droge w grafie miedzy wierzcholkiem "start", a wierzcholkiem "koniec"

Parameters

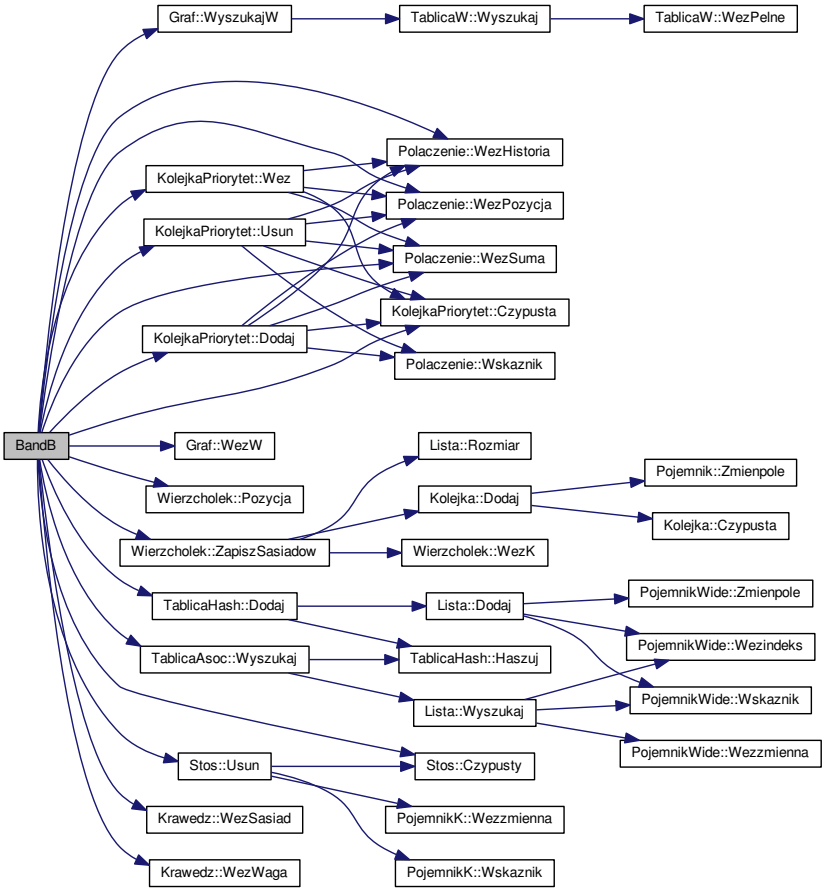
in	*start*	wierzcholek, z ktorego zaczynamy poszukiwania drogi
in	*meta*	wierzcholek, do ktorego droge wyznaczamy

Return values

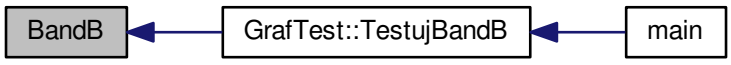
ilosc	rozwinietych krawedzi grafu przy wyznaczaniu najkrotszej drogi

Definition at line 14 of file BandB.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

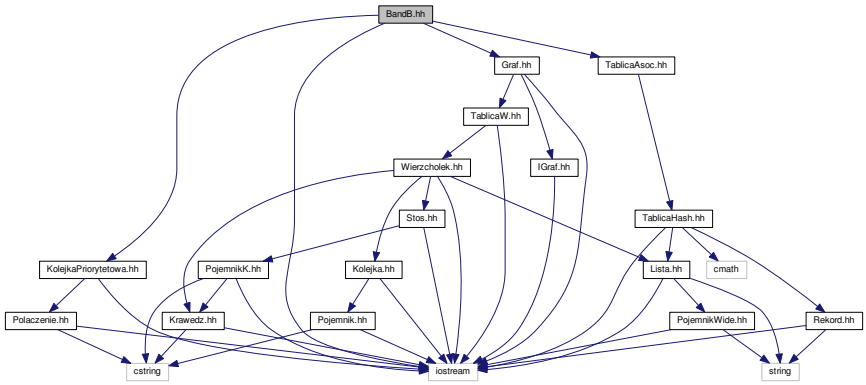


5.2 BandB.hh File Reference

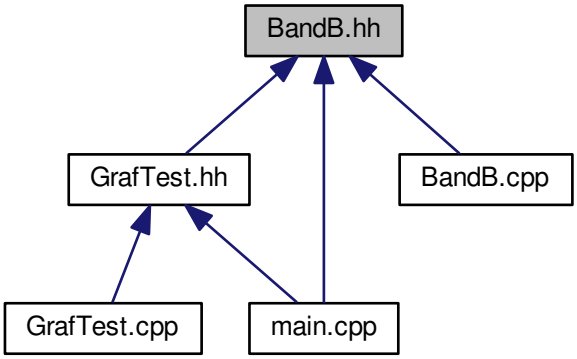
Implementacja funkcji B&B (Branch and bound)

```
#include <iostream>
#include "Graf.hh"
#include "TablicaAsoc.hh"
#include "KolejkaPriorytetowa.hh"
```

Include dependency graph for BandB.hh:



This graph shows which files directly or indirectly include this file:



Functions

- int BandB (Graf &graf, string start, string meta)

  Branch and Bound.

5.2.1 Function Documentation

5.2.1.1 int BandB (Graf & *graf,* string *start,* string *meta*)

Algorytm znajduje najkrotsza droge w grafie miedzy wierzcholkiem "start", a wierzcholkiem "koniec"

Parameters

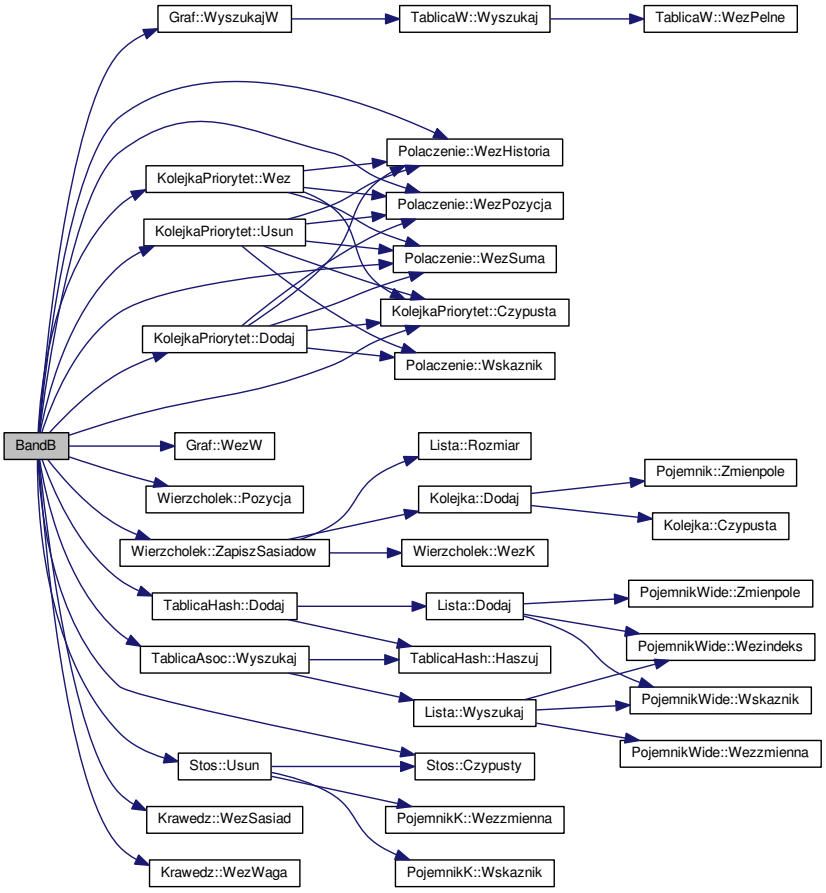
in	*start*	wierzcholek, z ktorego zaczynamy poszukiwania drogi
in	*meta*	wierzcholek, do ktorego droge wyznaczamy

Return values

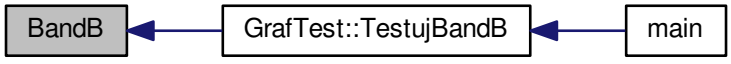
ilosc	rozwinietych krawedzi grafu przy wyznaczaniu najkrotszej drogi

Definition at line 14 of file BandB.cpp.

Here is the call graph for this function:



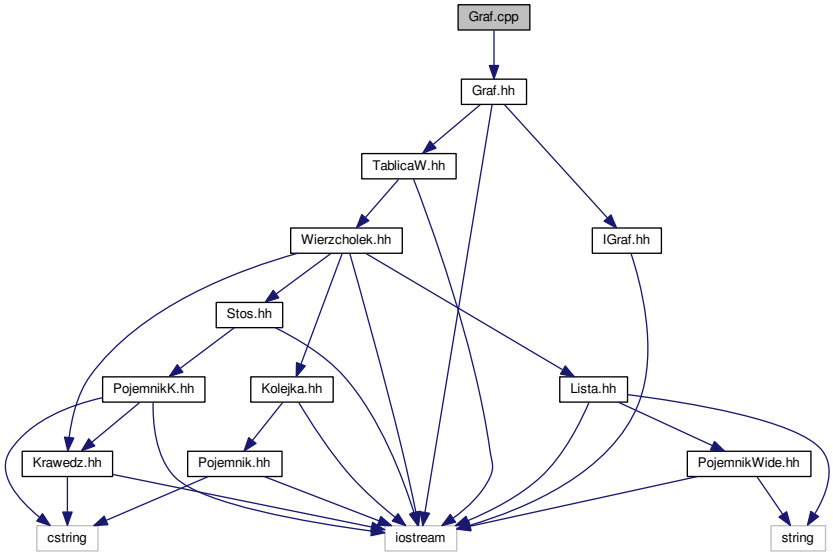
Here is the caller graph for this function:



5.3 Graf.cpp File Reference

```
#include "Graf.hh"
```

Include dependency graph for Graf.cpp:

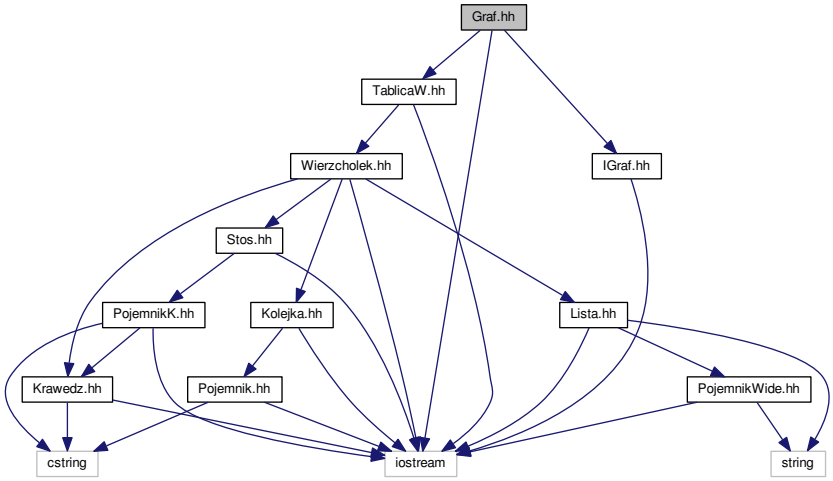


5.4 Graf.hh File Reference

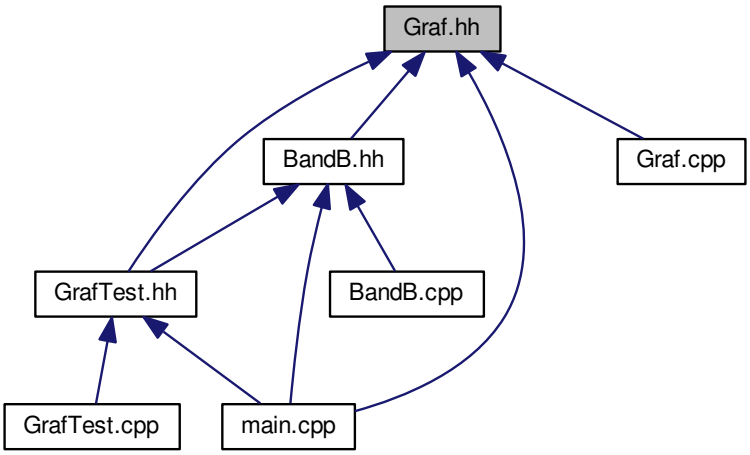
Implementacja grafu za pomoca listy sasiedztwa.

```
#include <iostream>
#include "TablicaW.hh"
#include "IGraf.hh"
```

Include dependency graph for Graf.hh:



This graph shows which files directly or indirectly include this file:



Classes

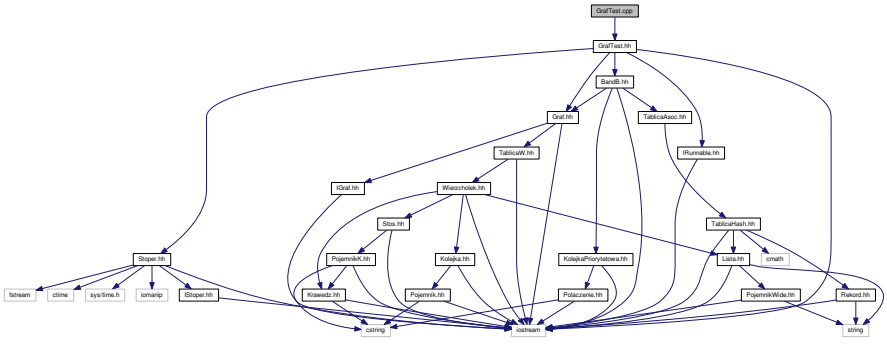
- class Graf

5.5 GrafTest.cpp File Reference

Definicja metod zwiazanych z "GrafTest".

```
#include "GrafTest.hh"
```

Include dependency graph for GrafTest.cpp:



Functions

- bool otworz_plik_zapis (string nazwapom, ofstream &StrmPlikowy)

  otwarcie pliku
- bool otworz_plik_odczyt (string nazwapom, fstream &StrmPlikowy)

  otwarcie pliku

5.5.1 Function Documentation

5.5.1.1 bool otworz_plik_odczyt (string *nazwapom*, fstream & *StrmPlikowy*)

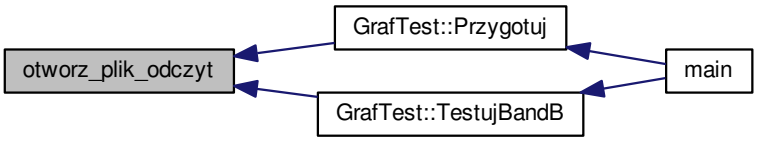
Otwiera plik i tworzy strumien do odczytu

Parameters

in	*nazwapom-*	nazwa pliku, ktory ma zostac otwarty
in	*StrmPlikowy-*	Zapisywany jest w nim strumien skad bedziemy odczytywac dane

Definition at line 39 of file GrafTest.cpp.

Here is the caller graph for this function:



5.5.1.2 bool otworz_plik_zapis (string *nazwapom*, ofstream & *StrmPlikowy*)

Otwiera plik i tworzy strumien do dopisywania

Parameters

in	*nazwapom-*	nazwa pliku, ktory ma zostac otwarty
in	*StrmPlikowy-*	Zapisywany jest w nim strumien gdzie mozemy dopisywac dane

Definition at line 17 of file GrafTest.cpp.

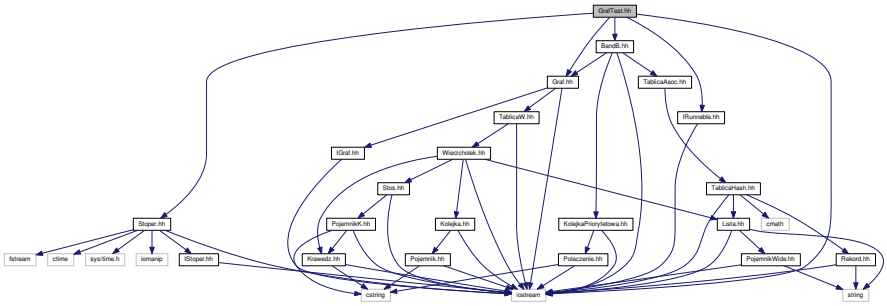
Here is the caller graph for this function:

5.6 GrafTest.hh File Reference

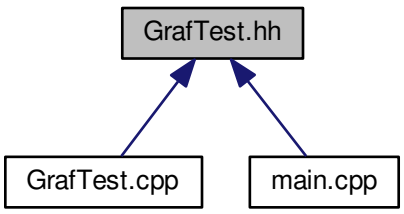
Implementacja klasy odpowiedzialnej za testowanie algorytmow DFS i BFS na grafie.

```
#include <iostream>
#include "Graf.hh"
#include "Stoper.hh"
#include "BandB.hh"
#include "IRunnable.hh"
```

Include dependency graph for GrafTest.hh:



This graph shows which files directly or indirectly include this file:



Classes

- class GrafTest

Functions

- bool otworz_plik_odczyt (string nazwapom, fstream &StrmPlikowy)

  otwarcie pliku
- bool otworz_plik_zapis (string nazwapom, fstream &StrmPlikowy)

5.6.1 Function Documentation

5.6.1.1 bool otworz_plik_odczyt (string *nazwapom,* fstream & *StrmPlikowy*)

Otwiera plik i tworzy strumien do odczytu

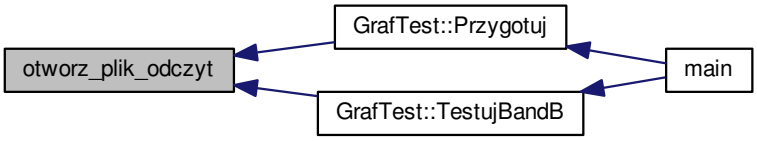
Parameters

in	*nazwapom-*	nazwa pliku, ktory ma zostac otwarty

in	*StrmPlikowy-*	Zapisywany jest w nim strumien skad bedziemy odczytywac dane

Definition at line 39 of file GrafTest.cpp.

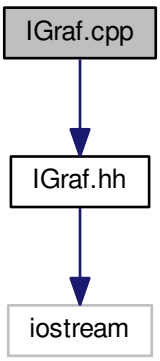
Here is the caller graph for this function:



5.6.1.2 bool otworz_plik_zapis (string *nazwapom*, fstream & *StrmPlikowy*)

5.7 IGraf.cpp File Reference

`#include "IGraf.hh"`
Include dependency graph for IGraf.cpp:

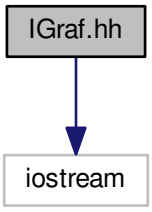


5.8 IGraf.hh File Reference

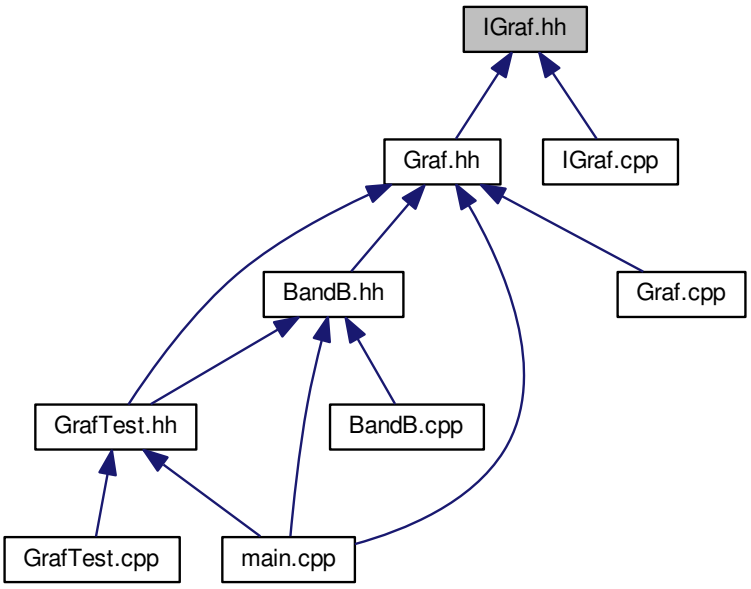
Interface Grafu.

```
#include <iostream>
```

Include dependency graph for IGraf.hh:



This graph shows which files directly or indirectly include this file:



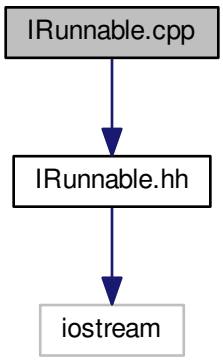
Classes

- class IGraf

5.9 IRunnable.cpp File Reference

```
#include "IRunnable.hh"
```

Include dependency graph for IRunnable.cpp:

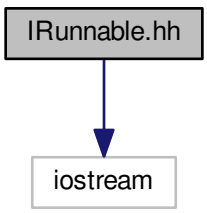


5.10 IRunnable.hh File Reference

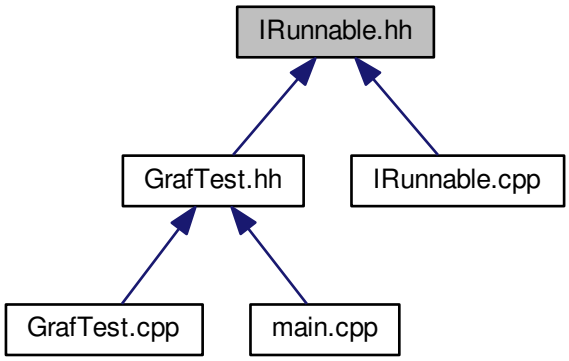
Interface testowania Grafu.

```
#include <iostream>
```

Include dependency graph for IRunnable.hh:



This graph shows which files directly or indirectly include this file:



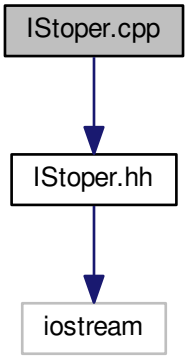
Classes

- class IRunnable

5.11 IStoper.cpp File Reference

```
#include "IStoper.hh"
```

Include dependency graph for IStoper.cpp:

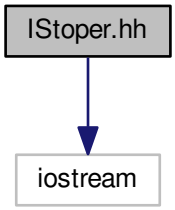


5.12 IStoper.hh File Reference

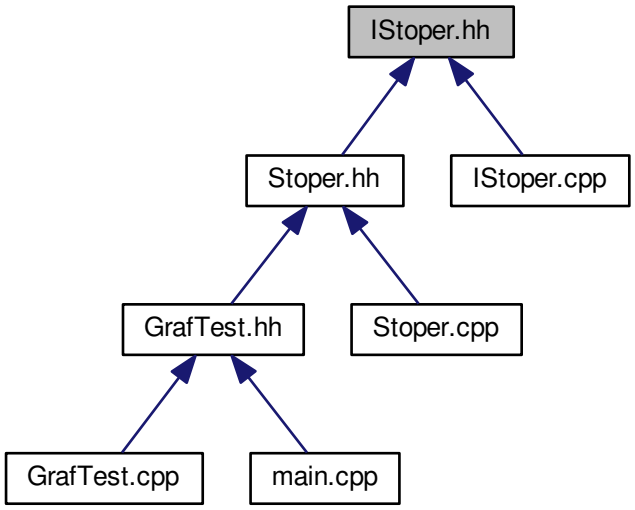
Interface Stoper.

```
#include <iostream>
```

Include dependency graph for IStoper.hh:



This graph shows which files directly or indirectly include this file:



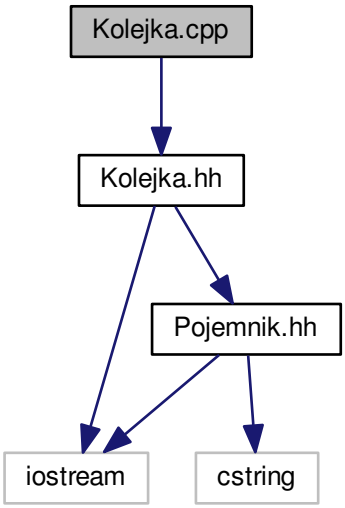
Classes

- class IStoper

5.13 Kolejka.cpp File Reference

Definicja metod ADT- Kolejka.

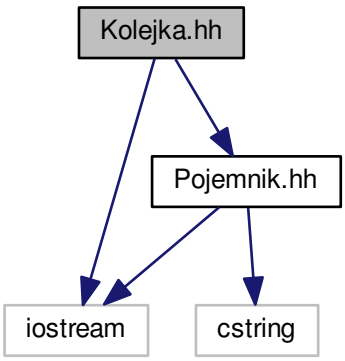
```
#include "Kolejka.hh"
```
Include dependency graph for Kolejka.cpp:



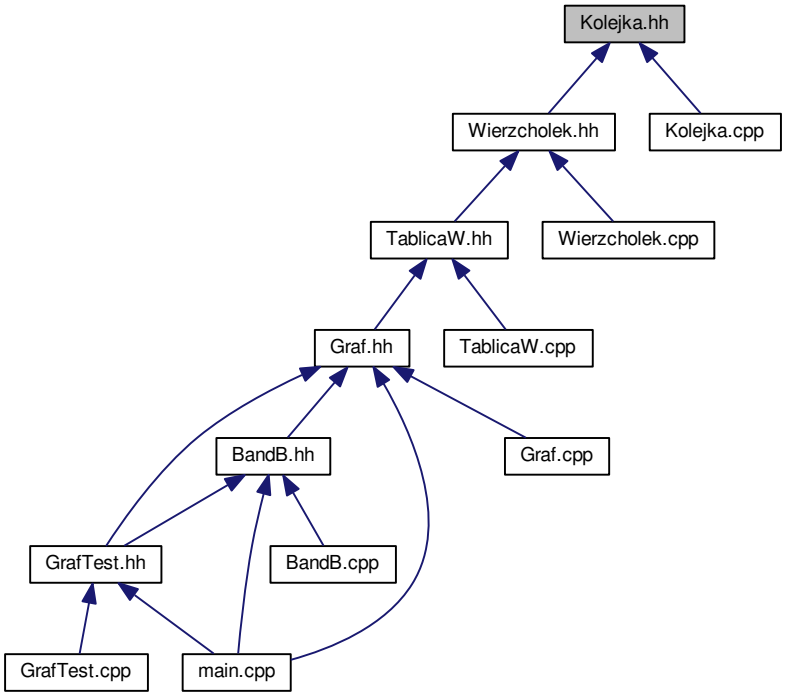
5.14 Kolejka.hh File Reference

implementacja abstrakcyjnego typu danych - Kolejka

```
#include "Pojemnik.hh"
#include <iostream>
```
Include dependency graph for Kolejka.hh:



This graph shows which files directly or indirectly include this file:



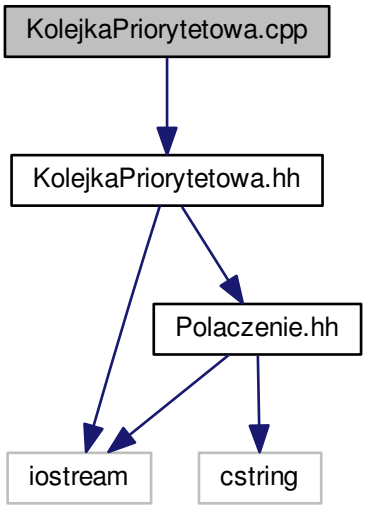
Classes

- class Kolejka

5.15 KolejkaPriorytetowa.cpp File Reference

Definicja metod ADT- KolejkaPriorytetowa.

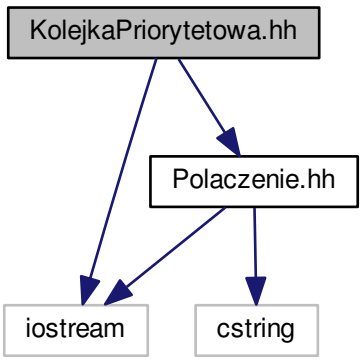
```
#include "KolejkaPriorytetowa.hh"
```
Include dependency graph for KolejkaPriorytetowa.cpp:



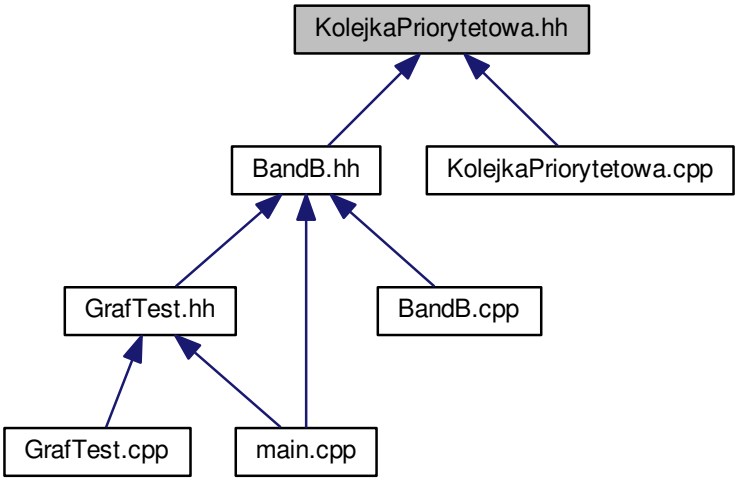
5.16 KolejkaPriorytetowa.hh File Reference

implementacja abstrakcyjnego typu danych - KolejkaPriorytetowa

```
#include "Polaczenie.hh"
#include <iostream>
```
Include dependency graph for KolejkaPriorytetowa.hh:



This graph shows which files directly or indirectly include this file:

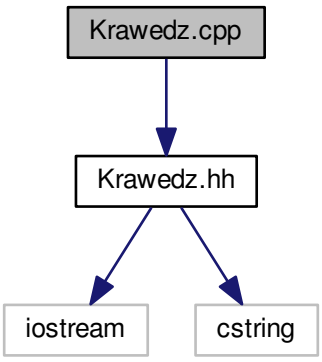


Classes

- class KolejkaPriorytet

5.17 Krawedz.cpp File Reference

`#include "Krawedz.hh"`
Include dependency graph for Krawedz.cpp:

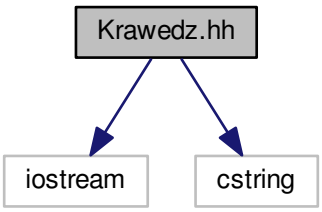


5.18 Krawedz.hh File Reference

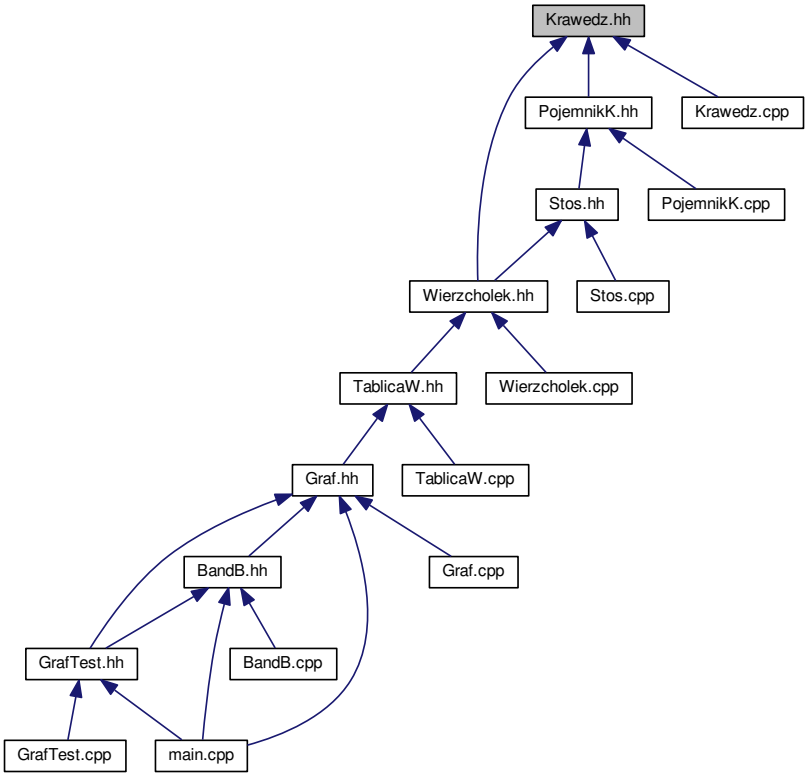
Implementacja krawedzi grafu.

```
#include <iostream>
#include <cstring>
```

Include dependency graph for Krawedz.hh:



This graph shows which files directly or indirectly include this file:

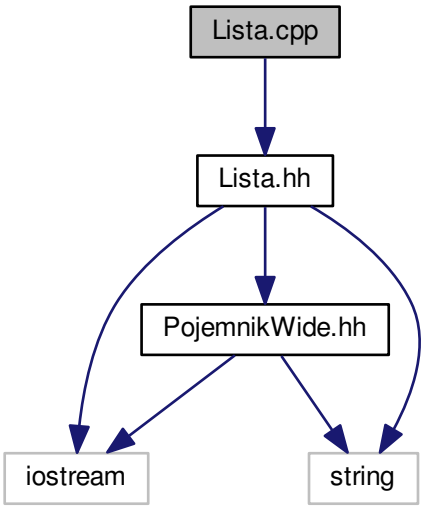


Classes

- class Krawedz

5.19 Lista.cpp File Reference

```
#include "Lista.hh"
```
Include dependency graph for Lista.cpp:

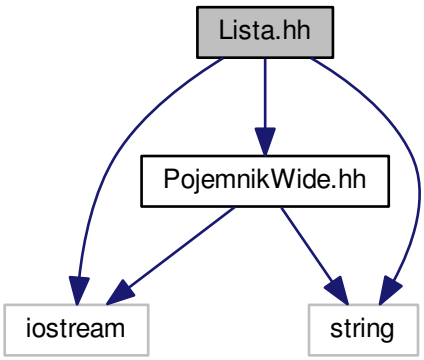


5.20 Lista.hh File Reference

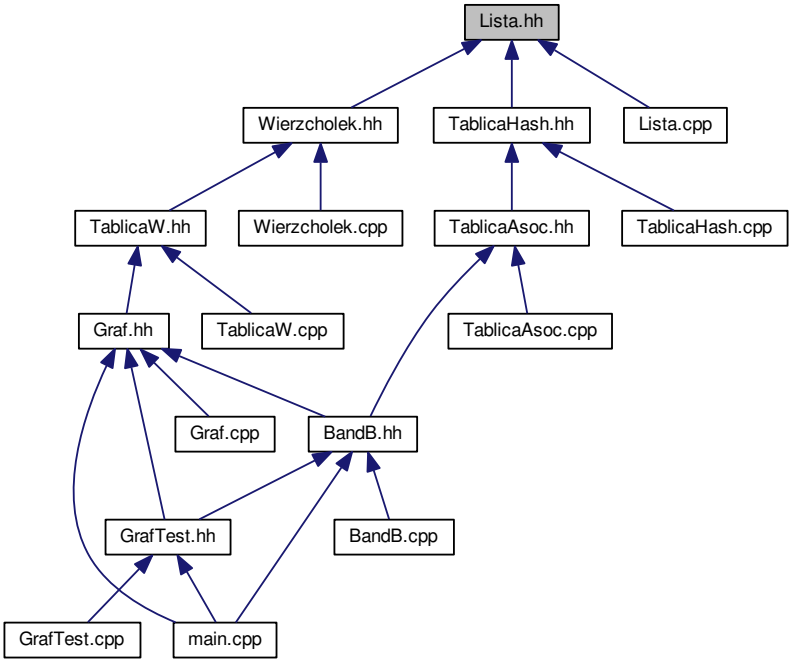
interface abstrakcyjnego typu danych - Lista

```
#include <iostream>
#include "PojemnikWide.hh"
#include <string>
```

Include dependency graph for Lista.hh:



This graph shows which files directly or indirectly include this file:



Classes

- class Lista< typ >

5.20.1 Detailed Description

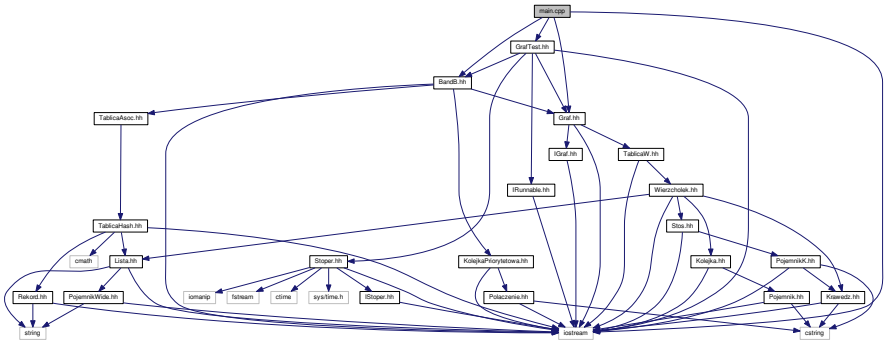
Elementy do Listy mozesz dodawac lub usuwac dowolnie czyli na poczatku, koncu badz wewnatrz listy

Definition in file Lista.hh.

5.21 main.cpp File Reference

```
#include <iostream>
#include "Graf.hh"
#include "BandB.hh"
#include "GrafTest.hh"
```

Include dependency graph for main.cpp:



Functions

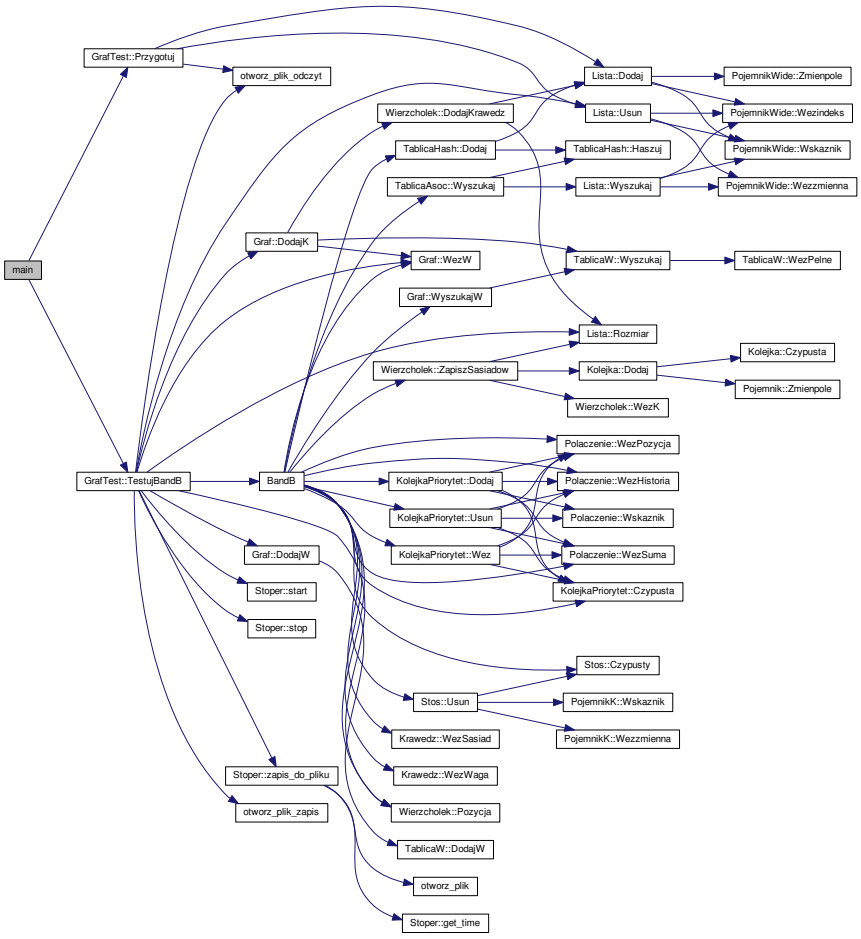
- int main ()

5.21.1 Function Documentation

5.21.1.1 int main ( )

Definition at line 9 of file main.cpp.

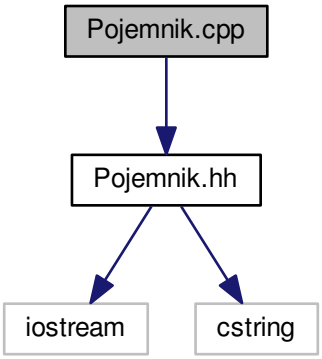
Here is the call graph for this function:



5.22 Pojemnik.cpp File Reference

Definicja metod pojedynczego elementu ADT (Kolejka, Stos)

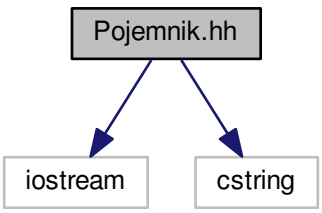
```
#include "Pojemnik.hh"
```
Include dependency graph for Pojemnik.cpp:



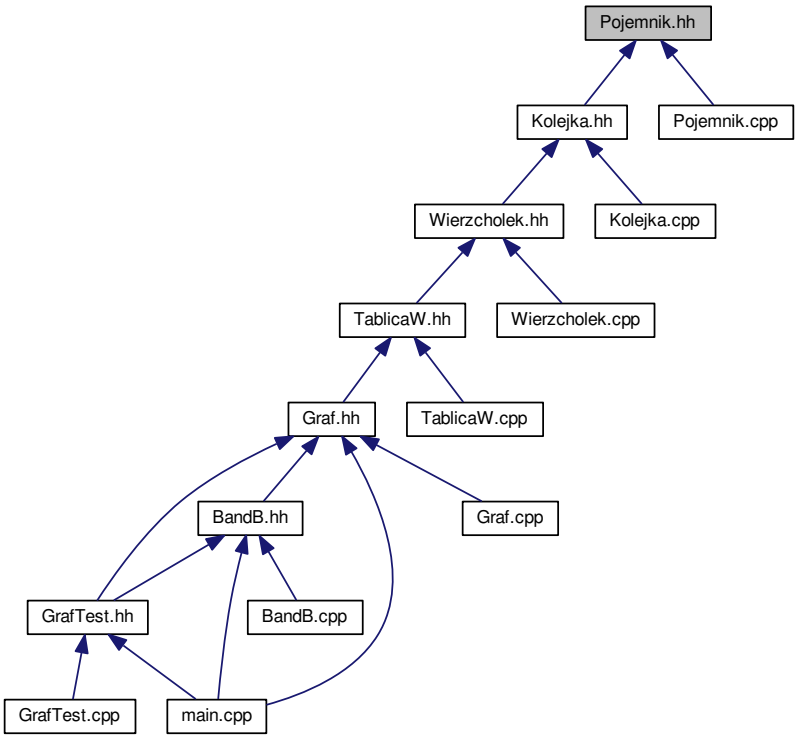
5.23 Pojemnik.hh File Reference

Pelni role pojedynczego elementu ADT (Kolejka, Stos)

```
#include <iostream>
#include <cstring>
```
Include dependency graph for Pojemnik.hh:



This graph shows which files directly or indirectly include this file:



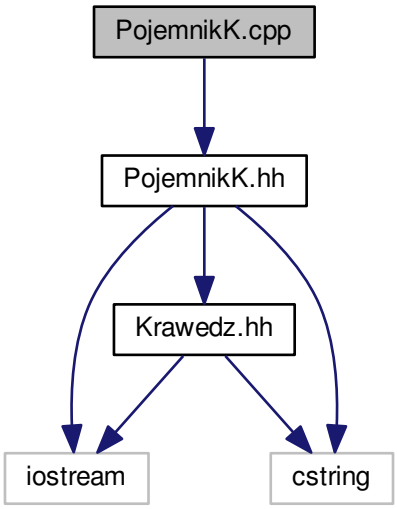
Classes

- class Pojemnik

5.24 PojemnikK.cpp File Reference

Definicja metod pojedynczego elementu ADT (Kolejka, Stos)

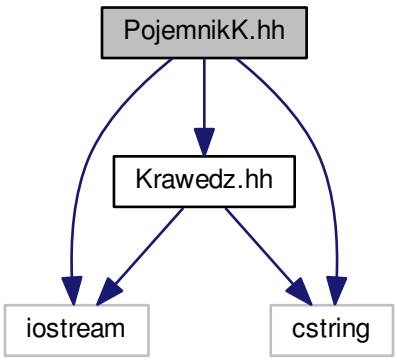
```
#include "PojemnikK.hh"
```
Include dependency graph for PojemnikK.cpp:



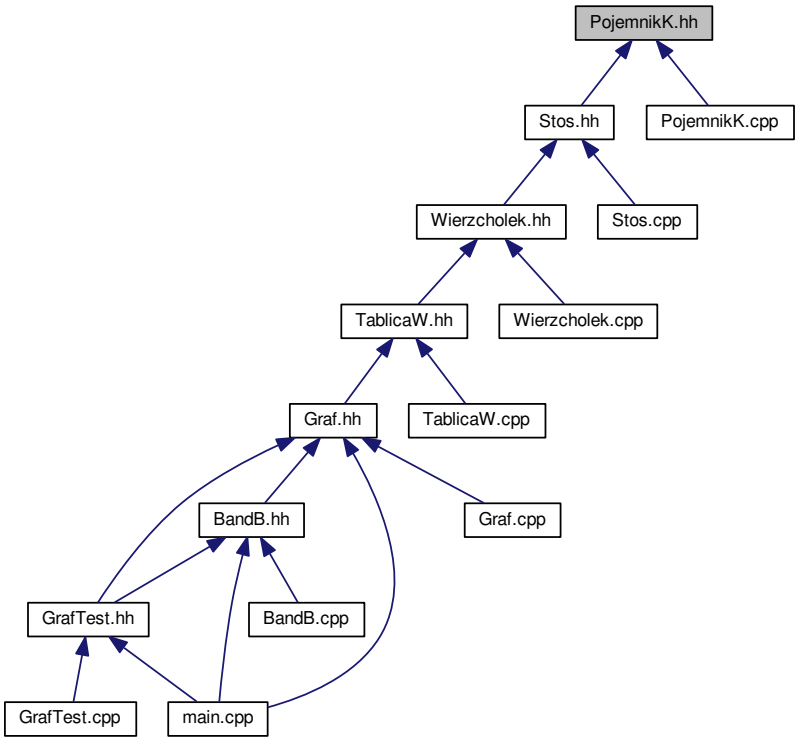
5.25 PojemnikK.hh File Reference

Pelni role pojedynczego elementu ADT (Kolejka, Stos)

```
#include <iostream>
#include <cstring>
#include <Krawedz.hh>
```
Include dependency graph for PojemnikK.hh:



This graph shows which files directly or indirectly include this file:



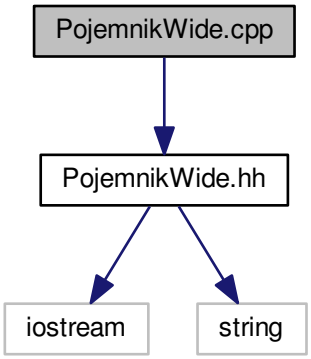
Classes

- class PojemnikK

5.26 PojemnikWide.cpp File Reference

Definicje metod pojedynczego elementu ADT (Lista)

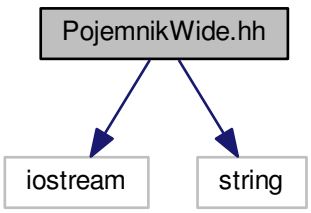
```
#include "PojemnikWide.hh"
```
Include dependency graph for PojemnikWide.cpp:



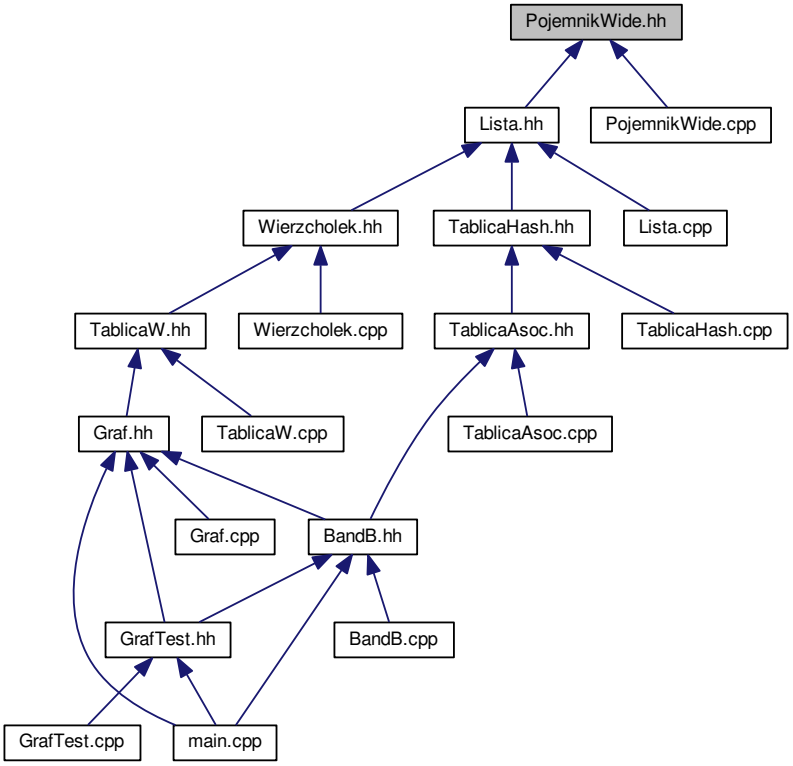
5.27 PojemnikWide.hh File Reference

Pelni role pojedynczego elementu ADT (Lista)

```
#include <iostream>
#include <string>
```
Include dependency graph for PojemnikWide.hh:



This graph shows which files directly or indirectly include this file:



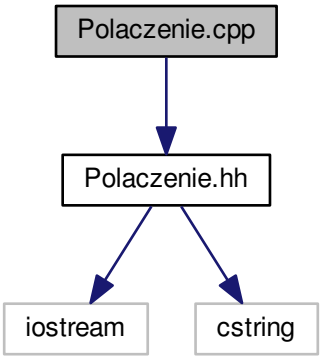
Classes

- class PojemnikWide< typ >

5.28 Polaczenie.cpp File Reference

Definicja metod pojedynczego elementu Kolejki priorytetowej.

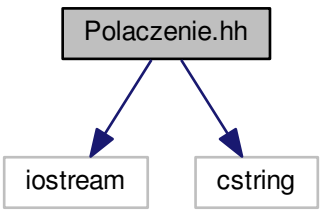
```
#include "Polaczenie.hh"
```
Include dependency graph for Polaczenie.cpp:



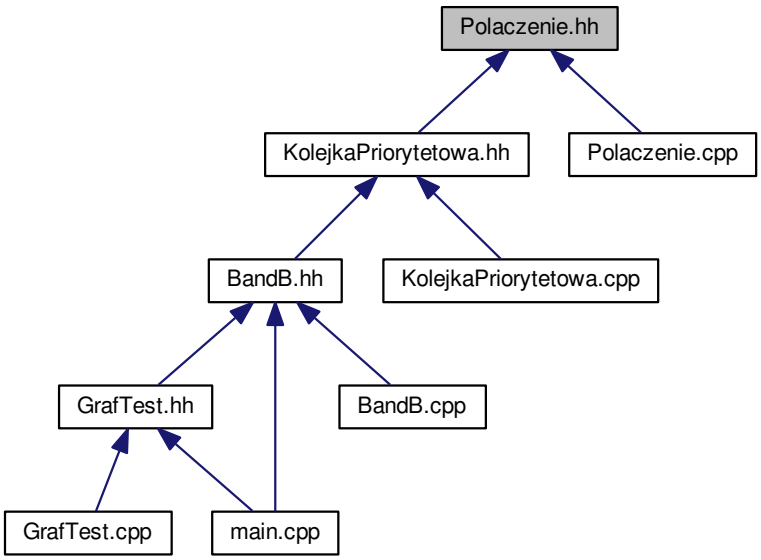
5.29 Polaczenie.hh File Reference

Pelni role pojedynczego elementu Kolejki Priorytetowej.

```
#include <iostream>
#include <cstring>
```
Include dependency graph for Polaczenie.hh:



This graph shows which files directly or indirectly include this file:



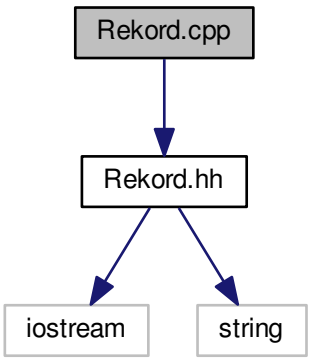
Classes

- class Polaczenie

5.30 Rekord.cpp File Reference

```
#include "Rekord.hh"
```

Include dependency graph for Rekord.cpp:



Functions

- bool operator== (Rekord istniejace, Rekord szukane)
- ostream & operator<< (ostream &strm, Rekord rek)

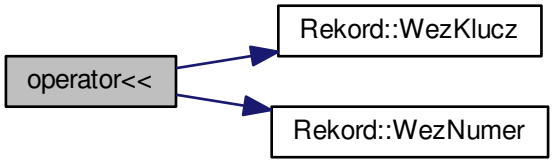
  Wyswietlanie rekordow.

5.30.1 Function Documentation

5.30.1.1 ostream& operator<< (ostream & *strm,* Rekord *rek*)

Definition at line 17 of file Rekord.cpp.

Here is the call graph for this function:

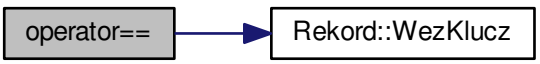


5.30.1.2 bool operator== (Rekord *istniejace,* Rekord *szukane*)

Przeciazenie operatora porownania dla dwoch obiektow typu "Rekord". Funkcja potrzebna do wyszukiwania rekordu w tablicy haszujacej

Definition at line 7 of file Rekord.cpp.

Here is the call graph for this function:

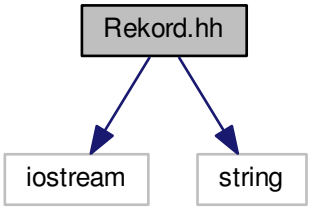


5.31 Rekord.hh File Reference

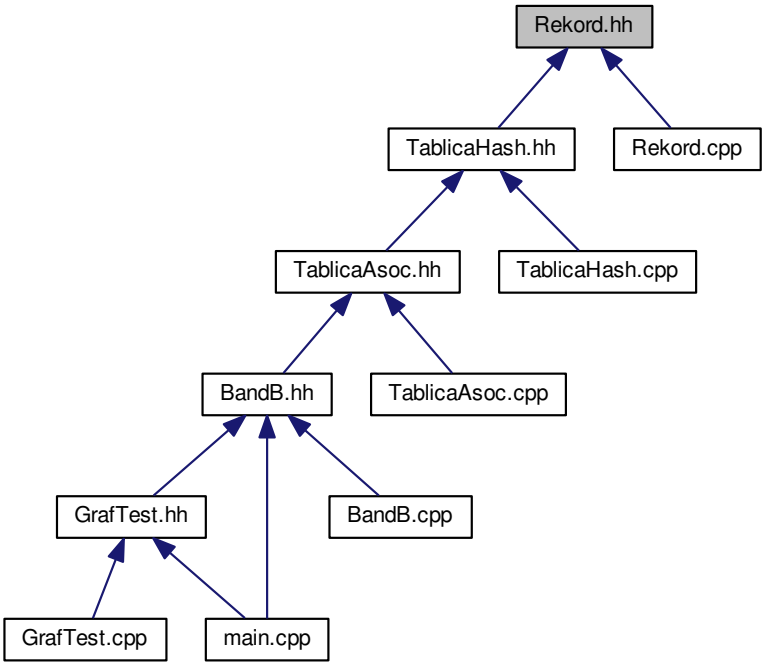
Implementacja pojedynczego rekordu "Ksiazki telefonicznej" (Tablica asocjacyjna)

```
#include <iostream>
#include <string>
```

Include dependency graph for Rekord.hh:



This graph shows which files directly or indirectly include this file:



Classes

- class Rekord

Functions

- bool operator== (Rekord istniejace, Rekord szukane)
- ostream & operator<< (ostream &strm, Rekord rek)

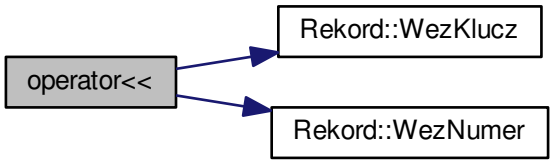
  Wyswietlanie rekordow.

5.31.1 Function Documentation

5.31.1.1 ostream& operator<< (ostream & *strm,* Rekord *rek*)

Definition at line 17 of file Rekord.cpp.

Here is the call graph for this function:

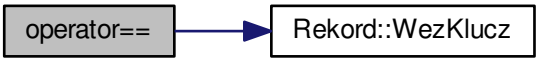


5.31.1.2 bool operator== (Rekord *istniejace,* Rekord *szukane*)

Przeciazenie operatora porownania dla dwoch obiektow typu "Rekord". Funkcja potrzebna do wyszukiwania rekordu w tablicy haszujacej

Definition at line 7 of file Rekord.cpp.

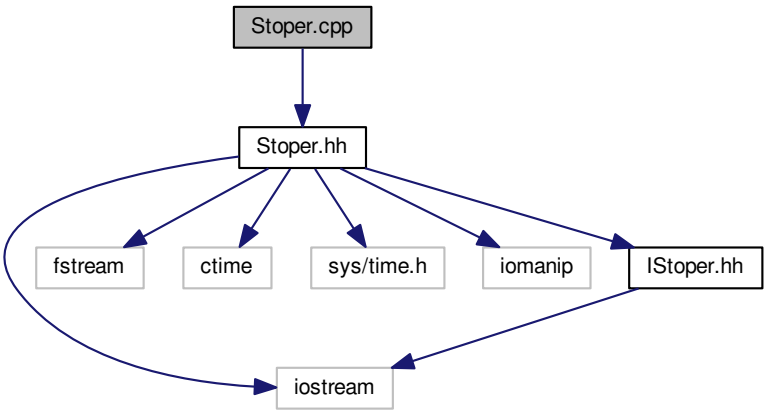
Here is the call graph for this function:



5.32 Stoper.cpp File Reference

```
#include "Stoper.hh"
```

Include dependency graph for Stoper.cpp:



Functions

- bool otworz_plik (string nazwapom, ofstream &StrmPlikowy)

  otwarcie pliku

5.32.1 Function Documentation

5.32.1.1 bool otworz_plik (string *nazwapom*, ofstream & *StrmPlikowy*)

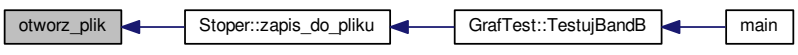
Otwiera plik i tworzy strumien do zapisywania UWAGA: PLIK OTWARTY W TRYBIE DOPISYWANIA

Parameters

in	*nazwapom*-	nazwa pliku, ktory ma zostac otwarty/utworzony
in	*StrmPlikowy*-	Zapisywany jest w nim strumien gdzie bedziemy zapisywac dane

Definition at line 23 of file Stoper.cpp.

Here is the caller graph for this function:

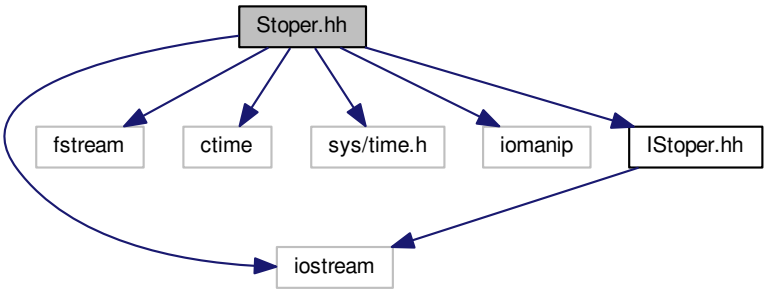


5.33 Stoper.hh File Reference

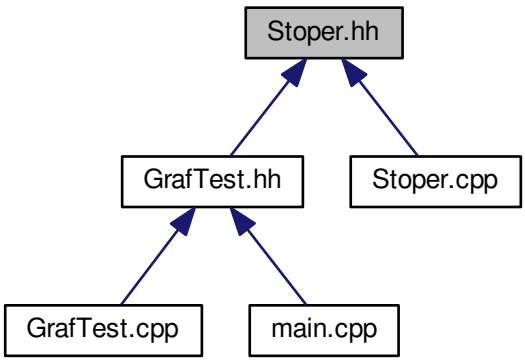
```
#include <iostream>
```

```
#include <fstream>
#include <ctime>
#include <sys/time.h>
#include <iomanip>
#include "IStoper.hh"
```

Include dependency graph for Stoper.hh:



This graph shows which files directly or indirectly include this file:



Classes

- class Stoper

Functions

- bool otworz_plik (string nazwapom, ofstream &StrmPlikowy)

  otwarcie pliku

5.33.1 Function Documentation

5.33.1.1 bool otworz_plik (string *nazwapom,* ofstream & *StrmPlikowy*)

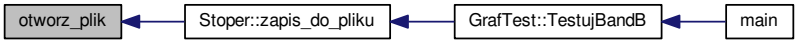
Otwiera plik i tworzy strumien do zapisywania UWAGA: PLIK OTWARTY W TRYBIE DOPISYWANIA

Parameters

in	*nazwapom-*	nazwa pliku, ktory ma zostac otwarty/utworzony
in	*StrmPlikowy-*	Zapisywany jest w nim strumien gdzie bedziemy zapisywac dane

Definition at line 23 of file Stoper.cpp.

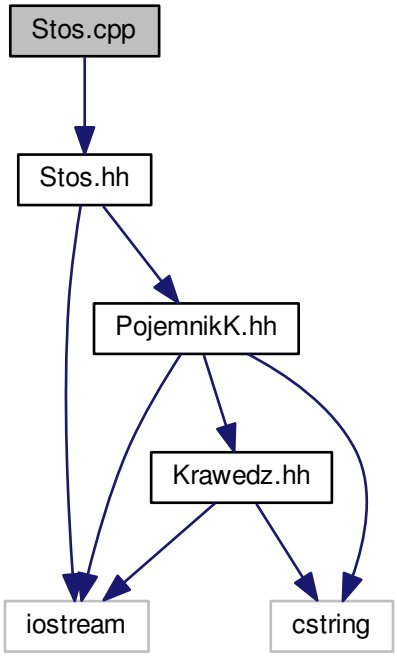
Here is the caller graph for this function:



5.34 Stos.cpp File Reference

Definicja metod interface'u ADT- Stos.

`#include "Stos.hh"`
Include dependency graph for Stos.cpp:

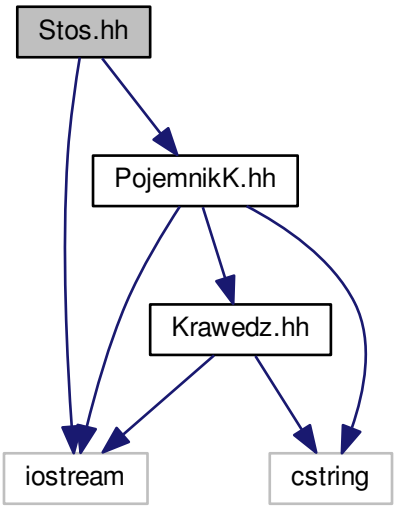


5.35 Stos.hh File Reference

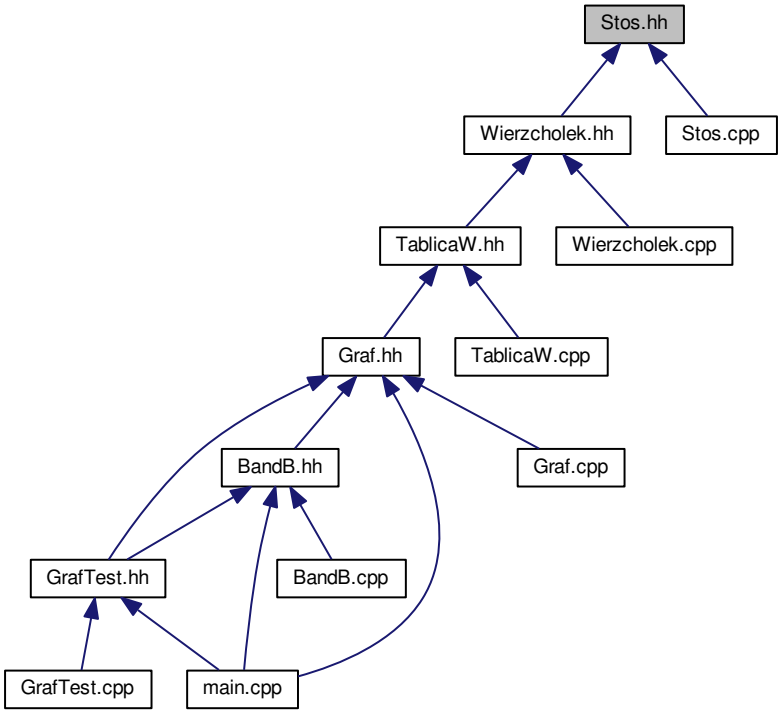
interface abstrakcyjnego typu danych - Stos

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

Include dependency graph for Stos.hh:



This graph shows which files directly or indirectly include this file:



Classes

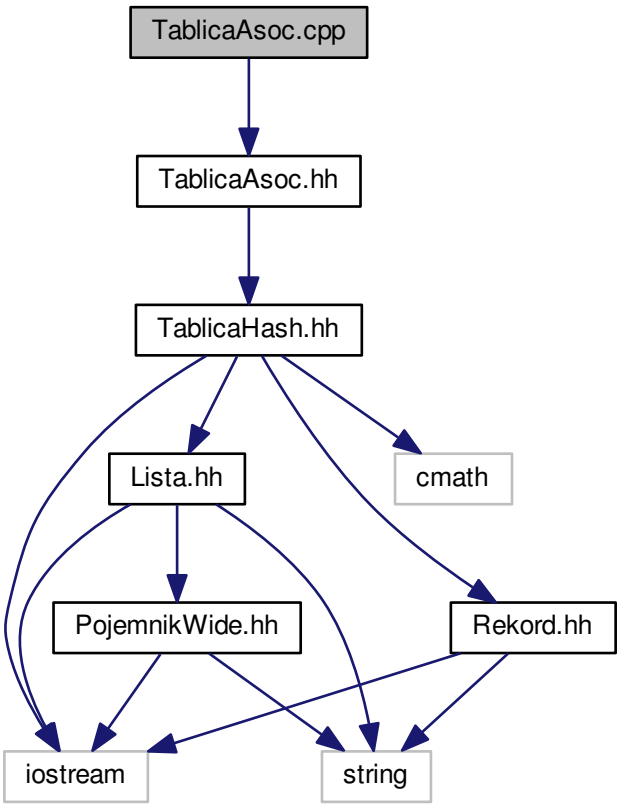
- class Stos

5.36 TablicaAsoc.cpp File Reference

Implementacja metod klasy TablicaAsoc.

```
#include "TablicaAsoc.hh"
```

Include dependency graph for TablicaAsoc.cpp:

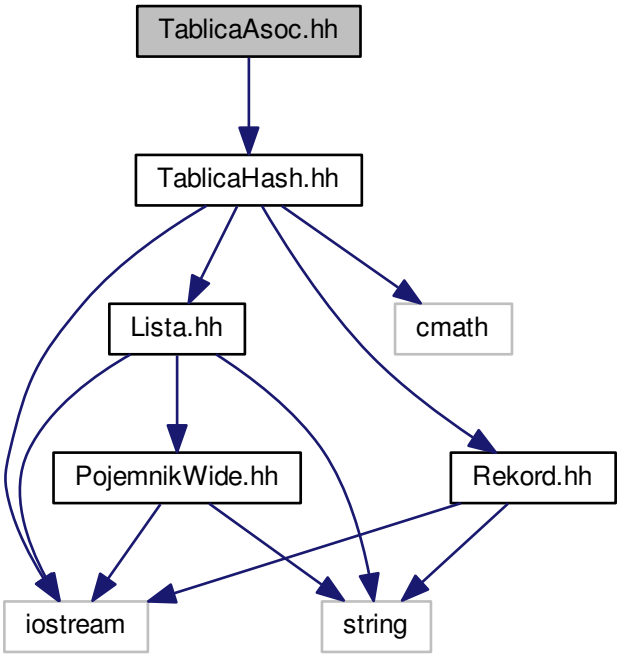


5.37 TablicaAsoc.hh File Reference

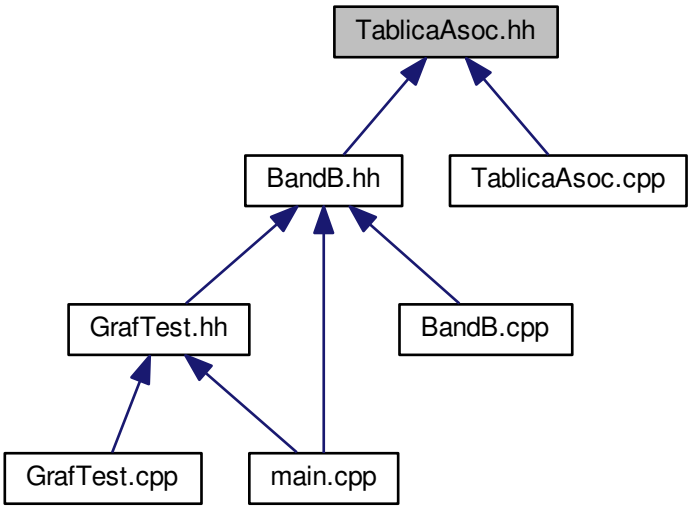
Tablica asocjacyjna.

```
#include "TablicaHash.hh"
```

Include dependency graph for TablicaAsoc.hh:



This graph shows which files directly or indirectly include this file:



Classes

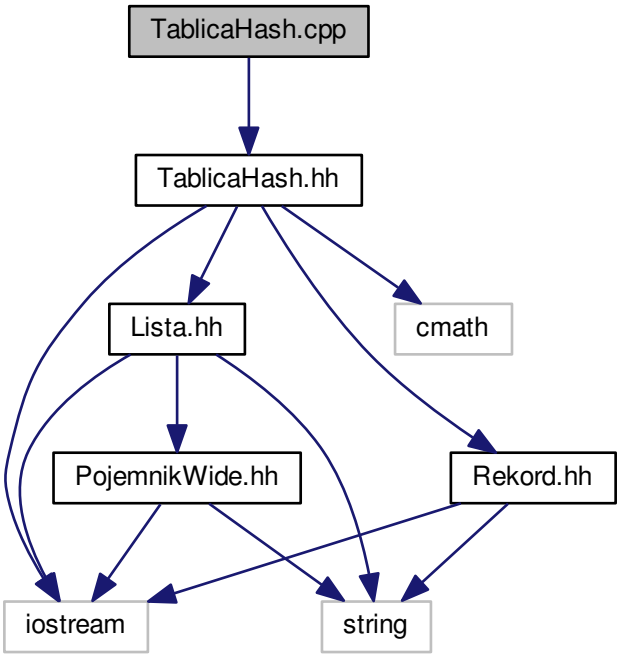
- class TablicaAsoc

5.38 TablicaHash.cpp File Reference

Implementacja metod tablicy hashujacej.

```
#include "TablicaHash.hh"
```

Include dependency graph for TablicaHash.cpp:

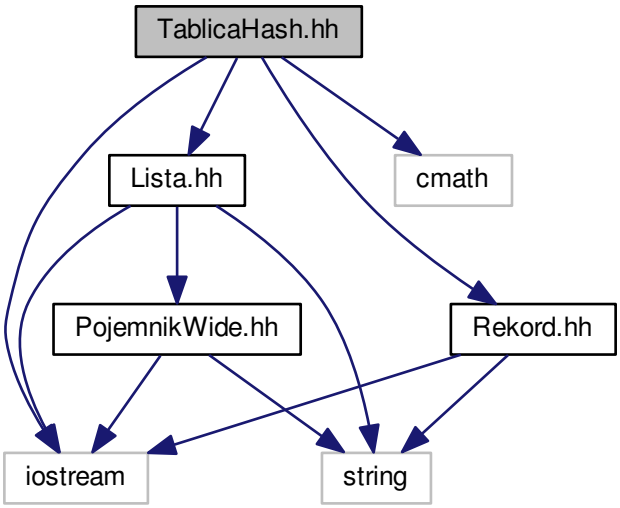


5.39 TablicaHash.hh File Reference

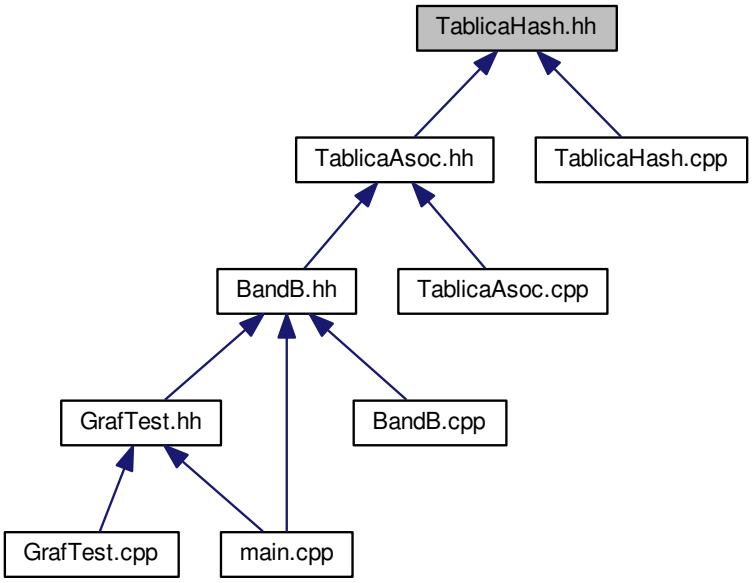
Tablica hashujaca (mieszajaca)

```
#include <iostream>
#include <cmath>
#include "Lista.hh"
#include "Rekord.hh"
```

Include dependency graph for TablicaHash.hh:



This graph shows which files directly or indirectly include this file:

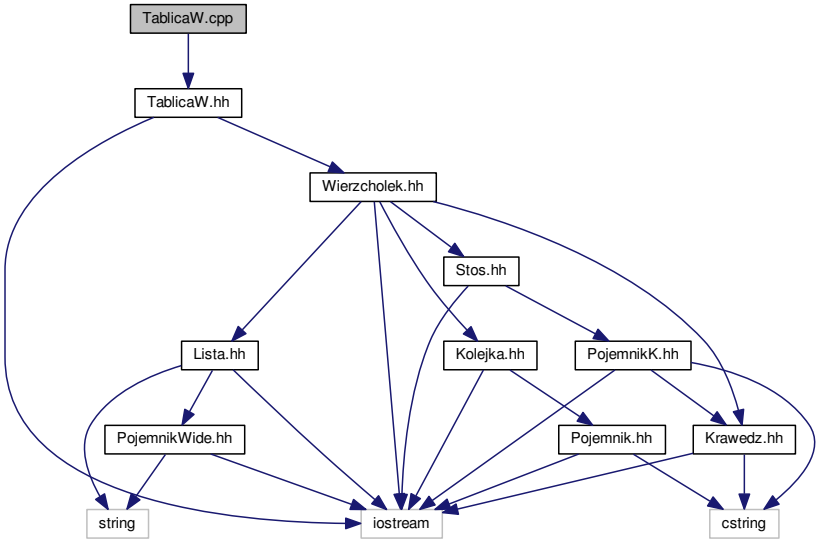


Classes

- class TablicaHash

5.40 TablicaW.cpp File Reference

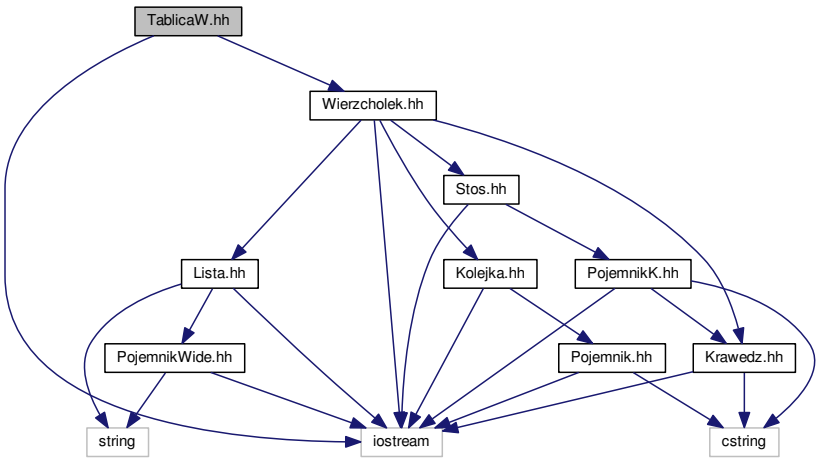
```
#include "TablicaW.hh"
```
Include dependency graph for TablicaW.cpp:



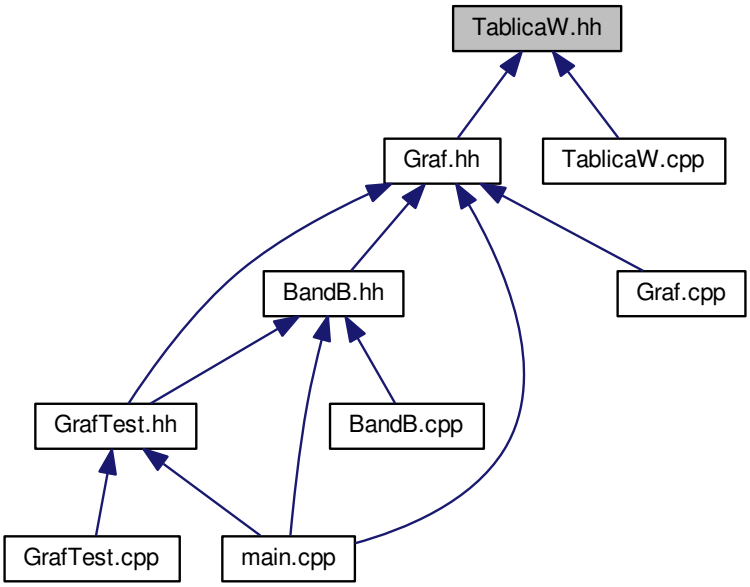
5.41 TablicaW.hh File Reference

Implementacja tablicy dynamicznej przechowujacej wierzcholki grafu.

```
#include <iostream>
#include "Wierzcholek.hh"
```
Include dependency graph for TablicaW.hh:



This graph shows which files directly or indirectly include this file:



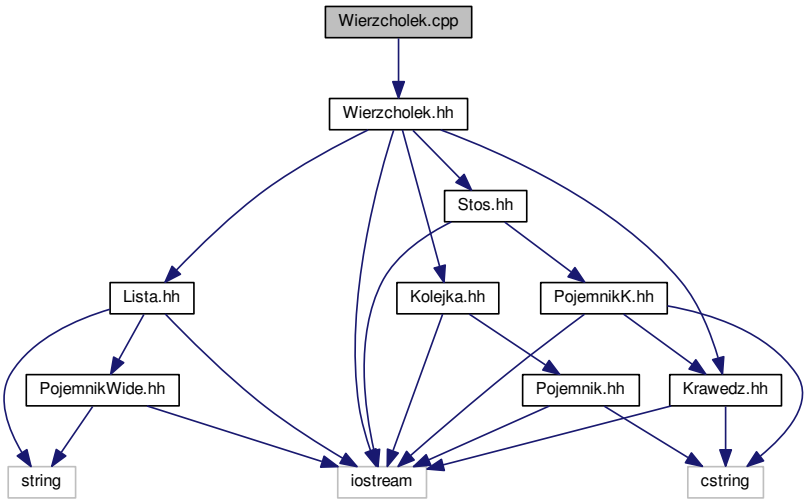
Classes

- class TablicaW

5.42 Wierzcholek.cpp File Reference

```
#include "Wierzcholek.hh"
```

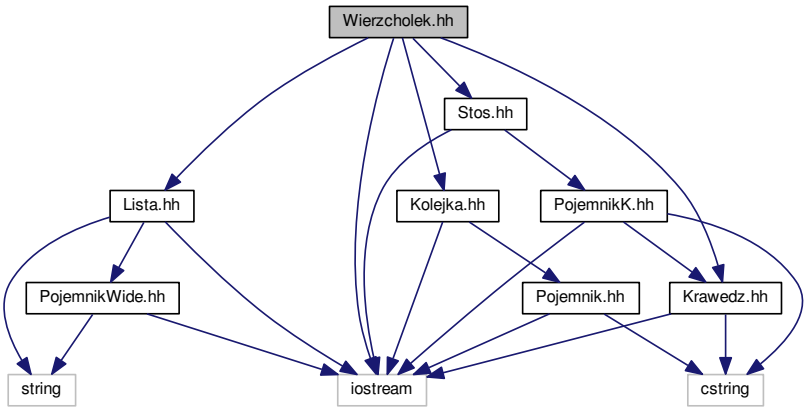
Include dependency graph for Wierzcholek.cpp:



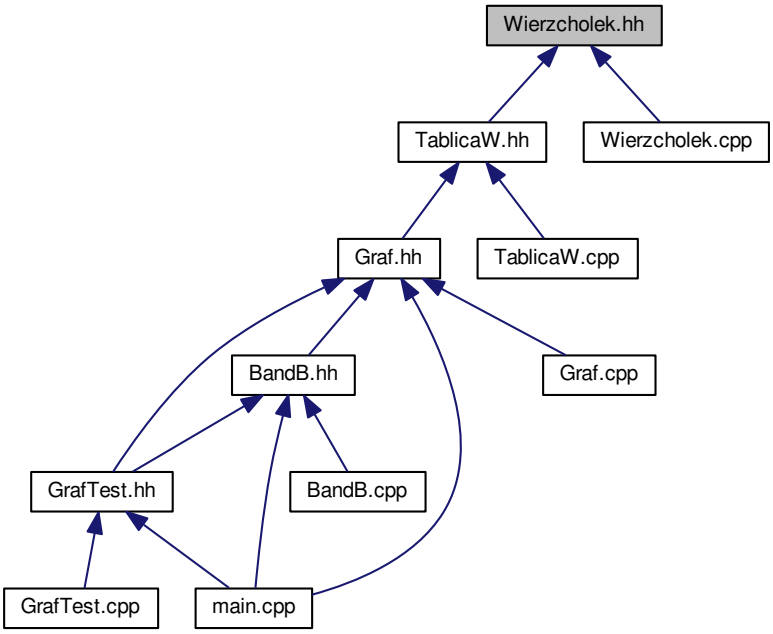
5.43 Wierzcholek.hh File Reference

```
#include <iostream>
#include "Lista.hh"
#include "Kolejka.hh"
#include "Krawedz.hh"
#include "Stos.hh"
```

Include dependency graph for Wierzcholek.hh:



This graph shows which files directly or indirectly include this file:



Classes

- class Wierzcholek