ADT

0.1

Generated by Doxygen 1.8.6

Sun May 8 2016 20:32:34

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:

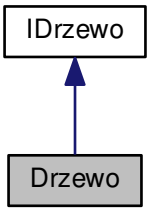
Chapter 4

Class Documentation

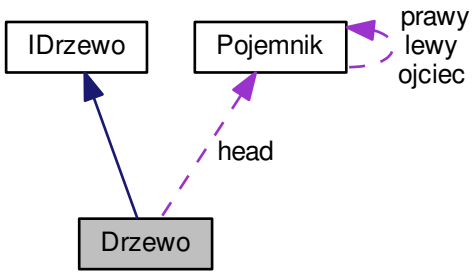
4.1 Drzewo Class Reference

`#include <Drzewo.hh>`

Inheritance diagram for Drzewo:



Collaboration diagram for Drzewo:



Public Member Functions

- Drzewo ()
- ~Drzewo ()
- bool Dodaj (int element)
- Pojemnik * Wyszukaj (int element)
- Pojemnik * DodajBezBalansu (int element)
- void LewaRotacja (Pojemnik *X)
- void PrawaRotacja (Pojemnik *X)
- string JakiJestem (Pojemnik *wezel)
- void Wyswietl (Pojemnik *start=NULL, string strona="korzen")

  Funkcja pomocnicza do sledzenia stanu drzewa.
- void Proba ()
- Pojemnik * Stryj (Pojemnik *Z)
- Pojemnik * Dziadek (Pojemnik *Z)

Private Member Functions

- void Usun (Pojemnik *pom)

Private Attributes

- Pojemnik * head =NULL

4.1.1 Detailed Description

Definition at line 15 of file Drzewo.hh.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 Drzewo::Drzewo ( ) [inline]

Definition at line 21 of file Drzewo.hh.

4.1.2.2 Drzewo::~Drzewo ( ) [inline]

Definition at line 22 of file Drzewo.hh.

4.1.3 Member Function Documentation

4.1.3.1 bool Drzewo::Dodaj (int *element*) [virtual]

Dodawanie elementu do drzewa czerwono-czarnego z zachowaniem balansu drzewa

Parameters

in	*element*-	element jaki ma byc przechowywany w drzewie

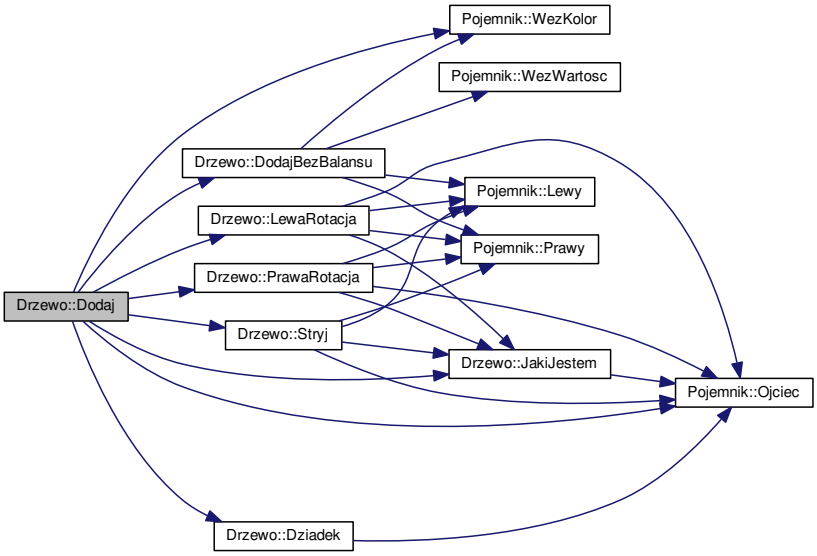
Return values

true	jesli operacja dodawania powiodla sie

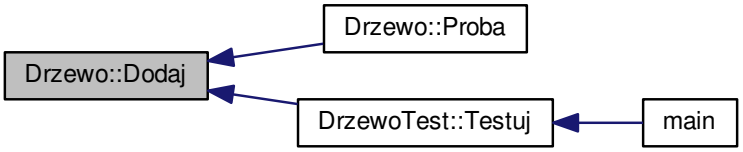
Implements IDrzewo.

Definition at line 203 of file Drzewo.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.3.2 Pojemnik * Drzewo::DodajBezBalansu (int *element*)

Dodawanie elementu tak jak dla zwyklego drzewa binarnego. Jest to metoda pomocnicza dla wlasciwej metody dodawania elementu do drzewa czerwono-czarnego

Parameters

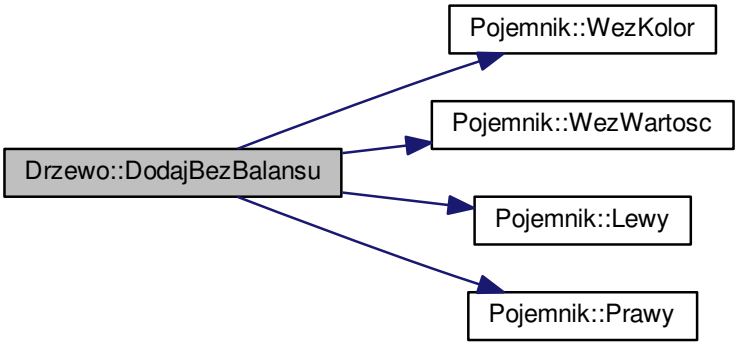
in	*element*-	element do przechowania w drzewie

Return values

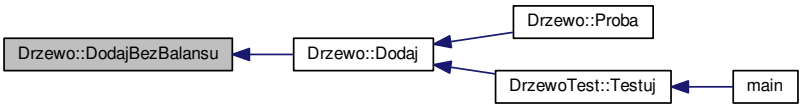
true-	jesli element zostal poprawnie dodany do drzewa

Definition at line 11 of file Drzewo.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



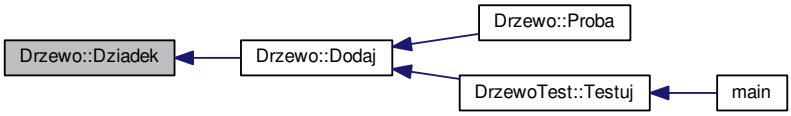
4.1.3.3 Pojemnik * Drzewo::Dziadek (Pojemnik * *Z*)

Daje nam dostep do dziadka wezla podanego jako parametr

Definition at line 157 of file Drzewo.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

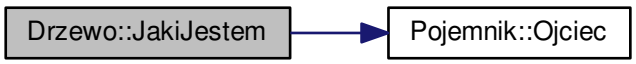


4.1.3.4 string Drzewo::JakiJestem (Pojemnik * *wezel*)

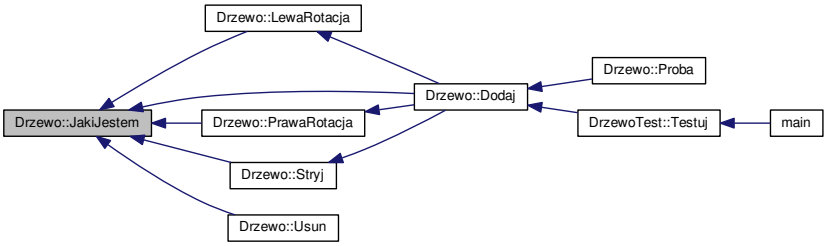
Sprawdza czy podany jako parametr wezel jest lewym czy prawym synem swego ojca

Definition at line 130 of file Drzewo.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.3.5 void Drzewo::LewaRotacja (Pojemnik * *X*)

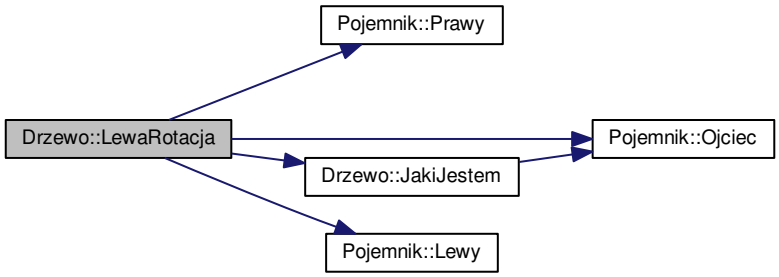
Lewa rotacja wzgledem wezla podanego jako parametr metody

Parameters

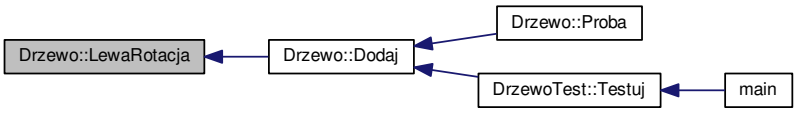
in	*X*-	wzgledem tego wezla nastepuje rotacja

Definition at line 58 of file Drzewo.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.3.6 void Drzewo::PrawaRotacja (Pojemnik * X)

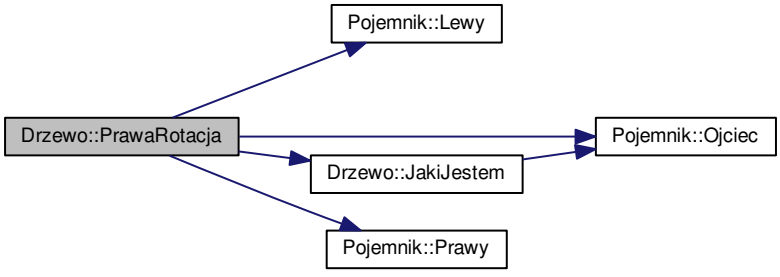
Prawa rotacja wzgledem wezla podanego jako parametr metody

Parameters

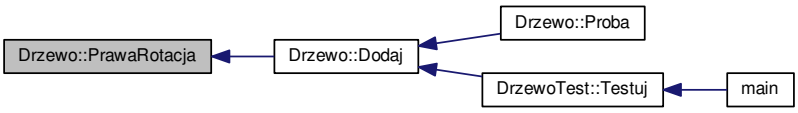
in	X-	wzgledem tego wezla nastepuje rotacja

Definition at line 95 of file Drzewo.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

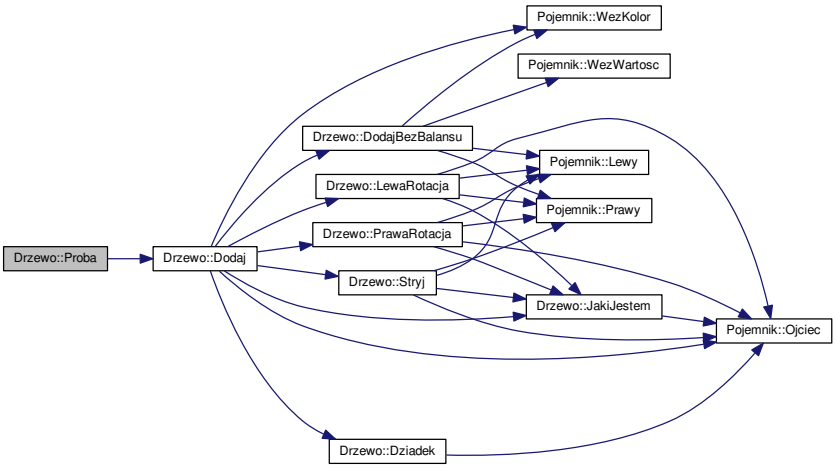


4.1.3.7 void Drzewo::Proba ( )

Funkcja pomocnicza do testowania algorytmu

Definition at line 166 of file Drzewo.cpp.

Here is the call graph for this function:

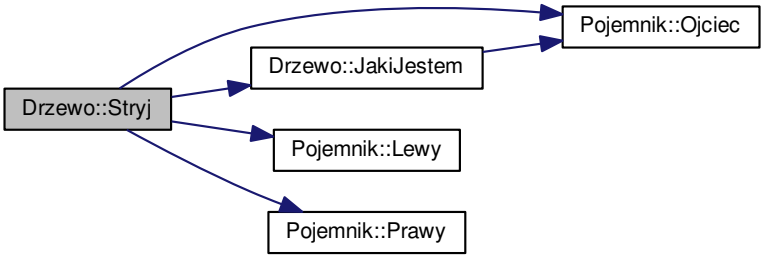


4.1.3.8 Pojemnik * Drzewo::Stryj (Pojemnik * Z)

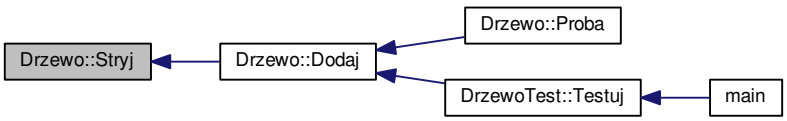
Daje nam dostep do stryja wezla podanego jako parametr

Definition at line 142 of file Drzewo.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

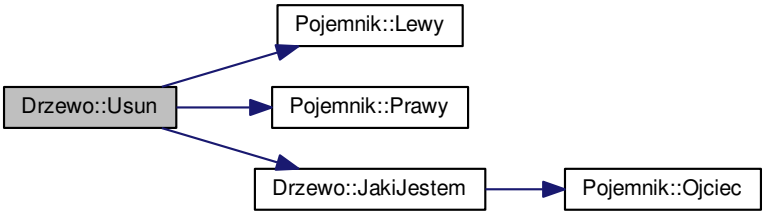


4.1.3.9 void Drzewo::Usun (Pojemnik * *pom*) [private]

Usuwa Wszystkie elementy z drzewa jesli w metodzie wywolamy Usun(head);

Definition at line 313 of file Drzewo.cpp.

Here is the call graph for this function:



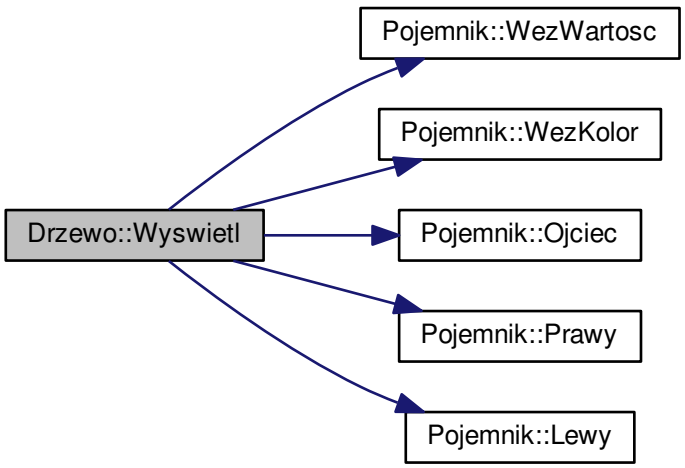
4.1.3.10 void Drzewo::Wyswietl (Pojemnik * *start* = NULL, string *strona* = "korzen")

Funkcja wypisujaca na standardowe wyjscie wszystkie wezly drzewa, informacje kto jest ich ojcem i informacje na temat przeplywu wywolan funkcji, a zarazem ksztaltu drzewa np. info, ze jest wezel "2", info o ojcu "2", a pozniej

info gdzie poszedlem z tego wezla "2" (poszedlem w prawo czyli wiemy , ze zaraz bedziemy widziec prawego syna wezla "2")

Definition at line 353 of file Drzewo.cpp.

Here is the call graph for this function:



4.1.3.11 Pojemnik * Drzewo::Wyszukaj (int *element*) [virtual]

Wyszukuje podany element w drzewie

Parameters

in	*element*-	poszukiwana wartosc

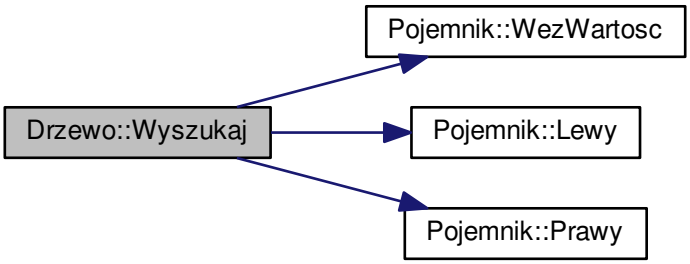
Return values

zwraca	adres do poszukiwanego elementu w drzewa w przypadku znalezienia elementu lub NULL w przypadku przeciwnym

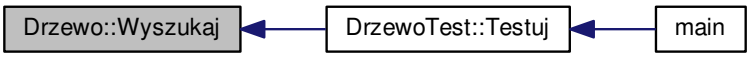
Implements IDrzewo.

Definition at line 389 of file Drzewo.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.4 Member Data Documentation

4.1.4.1 Pojemnik* Drzewo::head =NULL [private]

Definition at line 16 of file Drzewo.hh.

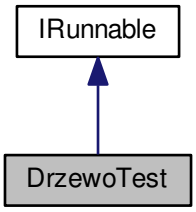
The documentation for this class was generated from the following files:

- Drzewo.hh
- Drzewo.cpp

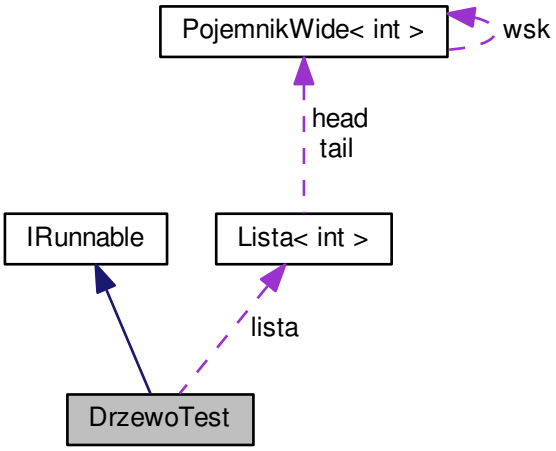
4.2 DrzewoTest Class Reference

`#include <DrzewoTest.hh>`

Inheritance diagram for DrzewoTest:



Collaboration diagram for DrzewoTest:



Public Member Functions

- bool Przygotuj (string nazwapliku)
- void Testuj ()

Private Attributes

- Lista< int > lista

4.2.1 Detailed Description

Definition at line 16 of file DrzewoTest.hh.

4.2.2 Member Function Documentation

4.2.2.1 bool DrzewoTest::Przygotuj (string *nazwapliku*) [virtual]

Odczytuje jaka ilosc elementow ma byc zapisana w drzewie i zapisuje odczytane wartosci na liscie. Wartosci odczytywae sa z pliku. Przykladowa zawartosc pliku: 10 100 1000

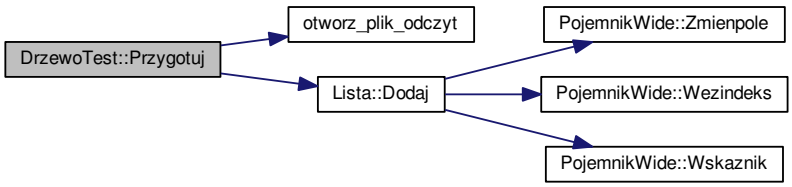
Parameters

in	*nazwapliku-*	nazwa pliku, w ktorum przechowywane sa dane na temat ilosci liczb do zapisania w drzewie

Implements IRunnable.

Definition at line 38 of file DrzewoTest.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

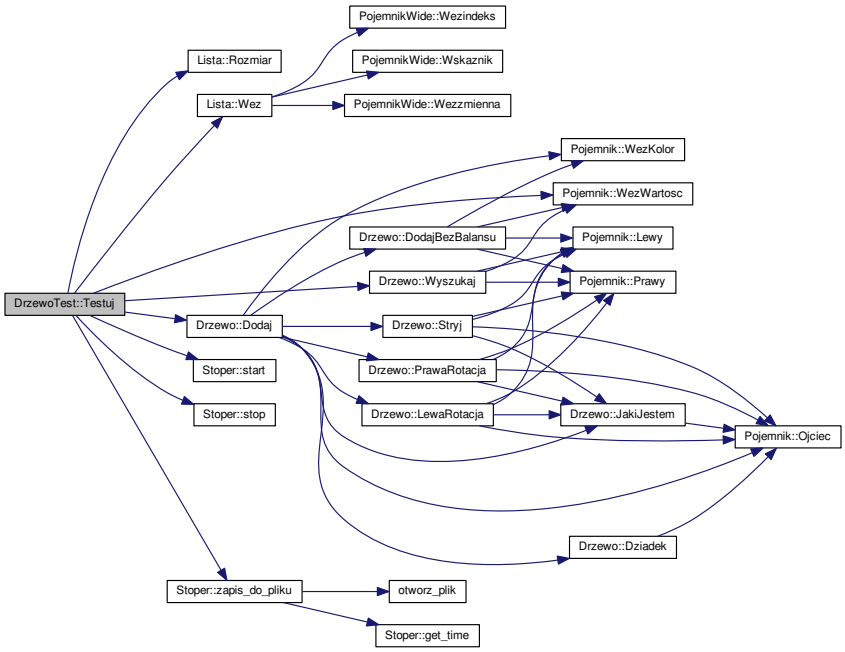
4.2.2.2 void DrzewoTest::Testuj ( ) [virtual]

Zapisuje odpowiednia ilosc elementow w drzewie (odpowiednia-przechowywana na liscie), wyszukuje podana wartosc w drzewie, mierzy czas przeszukiwania i zapisuje go do pliku "czasy.dat"

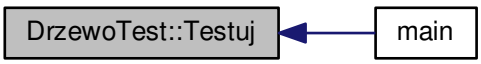
Implements IRunnable.

Definition at line 58 of file DrzewoTest.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



4.2.3 Member Data Documentation

4.2.3.1 Lista<int> DrzewoTest::lista [private]

Definition at line 17 of file DrzewoTest.hh.

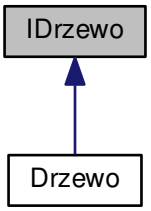
The documentation for this class was generated from the following files:

- DrzewoTest.hh
- DrzewoTest.cpp

4.3 IDrzewo Class Reference

`#include <IDrzewo.hh>`

Inheritance diagram for IDrzewo:



Public Member Functions

- virtual bool Dodaj (int element)=0
- virtual Pojemnik ∗ Wyszukaj (int element)=0

4.3.1 Detailed Description

Definition at line 14 of file IDrzewo.hh.

4.3.2 Member Function Documentation

4.3.2.1 virtual bool IDrzewo::Dodaj (int *element*) [pure virtual]

Implemented in Drzewo.

4.3.2.2 virtual Pojemnik∗ IDrzewo::Wyszukaj (int *element*) [pure virtual]

Implemented in Drzewo.

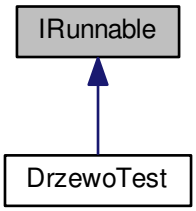
The documentation for this class was generated from the following file:

- IDrzewo.hh

4.4 IRunnable Class Reference

`#include <IRunnable.hh>`

Inheritance diagram for IRunnable:



Public Member Functions

- virtual bool Przygotuj (string nazwapliku)=0
- virtual void Testuj ()=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 *nazwapliku*) [pure virtual]

Implemented in DrzewoTest.

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

Implemented in DrzewoTest.

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

- IRunnable.hh

4.5 Lista< typ > Class Template Reference

```
#include <Lista.hh>
```

Public Member Functions

- ~Lista ()
- 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

4.5.1 Detailed Description

template<typename typ>class Lista< typ >

Definition at line 18 of file Lista.hh.

4.5.2 Constructor & Destructor Documentation

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

Destruktor - Usuwa wszystkie elementy z listy (na bazie metody "Oproznij")

Definition at line 286 of file Lista.hh.

4.5.3 Member Function Documentation

4.5.3.1 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.

4.5.3.2 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

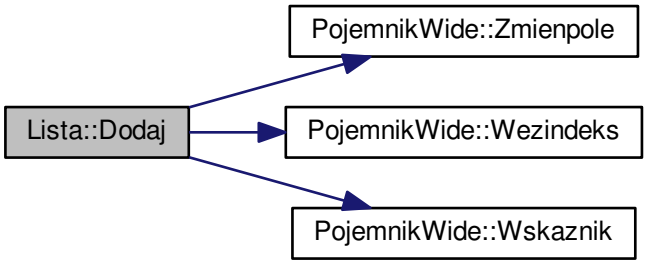
in	*elem*-	wartosc do przechowania
in	*index*-	indeks listy pod jakim bedzie przechowywany pojemnik ze zmienna

Return values

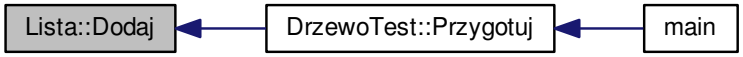
false-	gdy element ma byc wstawiony w nielogicznym miejscu, np-> wstawianie elementu o indeksie 100 kiedy lista ma aktualnie indeksy od 0 do 15
true-	gdy element wstawiono poprawnie do listy

Definition at line 53 of file Lista.hh.

Here is the call graph for this function:



Here is the caller graph for this function:



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

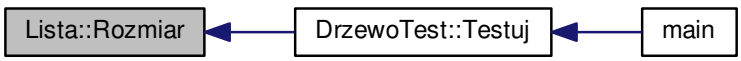
Usuwa wszystkie elementy z listy

Definition at line 228 of file Lista.hh.

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

Definition at line 27 of file Lista.hh.

Here is the caller graph for this function:



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

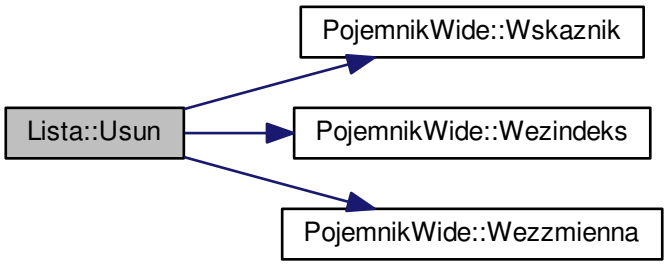
Usuwa element z Listy o zadanym indeksie i zwraca wartosc, ktora przechowywal

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:



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

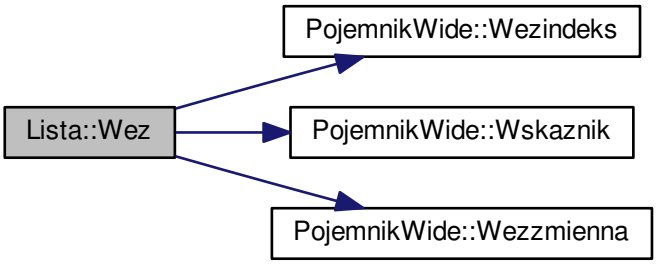
Zwraca wartość elementu o zadanym indeksie

Parameters

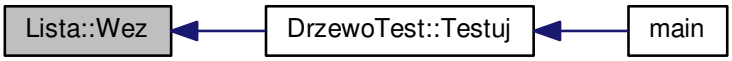
in	*ind-*	indeks poszukiwanego elementu

Definition at line 118 of file Lista.hh.

Here is the call graph for this function:



Here is the caller graph for this function:

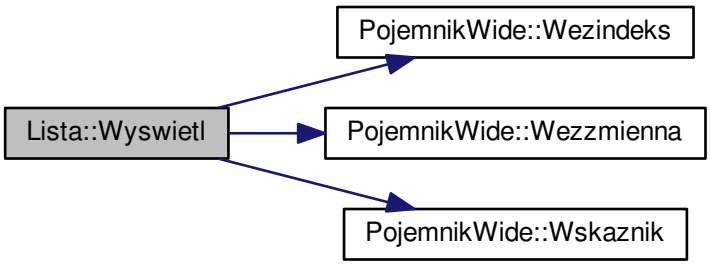


4.5.3.7 template<typename typ > void Lista< typ >::Wyswietl ( )

Wyswietla zawartosc listy na standardowe wyjscie

Definition at line 242 of file Lista.hh.

Here is the call graph for this function:



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

Wyszukuje podany wyraz wsrod elementow listy

Parameters

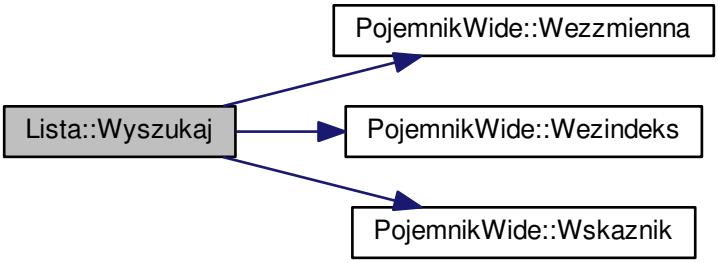
in	*szukane*-	szukany wyraz

Return values

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

Definition at line 269 of file Lista.hh.

Here is the call graph for this function:



4.5.4 Member Data Documentation

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

Definition at line 19 of file Lista.hh.

4.5.4.2 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

4.6 Pojemnik Class Reference

`#include <Pojemnik.hh>`

Collaboration diagram for Pojemnik:

Public Member Functions

- int & WezWartosc ()
- string & WezKolor ()
- Pojemnik * Ojciec ()

- Pojemnik * Lewy ()
- Pojemnik * Prawy ()

Public Attributes

- Pojemnik * ojciec =NULL
- Pojemnik * lewy =NULL
- Pojemnik * prawy =NULL

Private Attributes

- int wartosc =0
- string kolor ="R"

4.6.1 Detailed Description

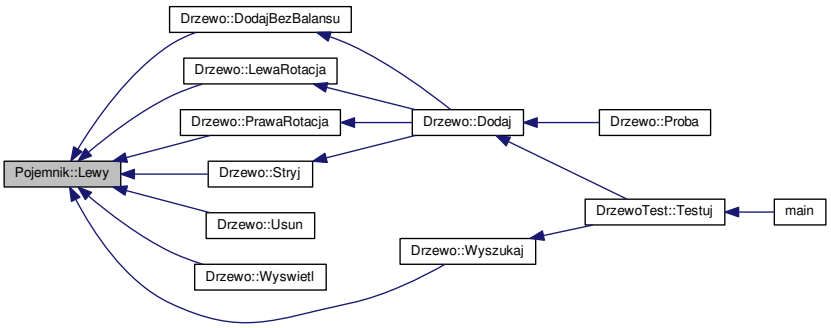
Definition at line 14 of file Pojemnik.hh.

4.6.2 Member Function Documentation

4.6.2.1 Pojemnik* Pojemnik::Lewy ( ) [inline]

Definition at line 26 of file Pojemnik.hh.

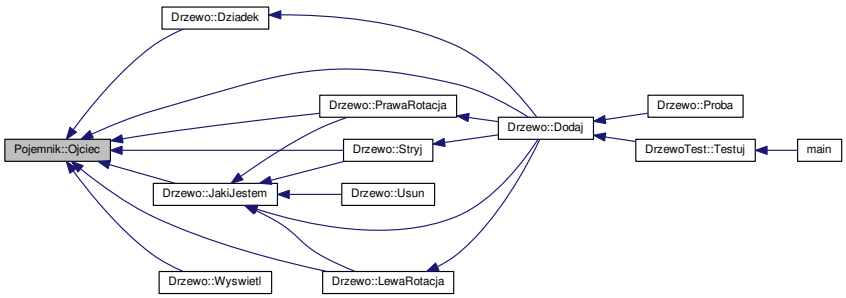
Here is the caller graph for this function:



4.6.2.2 Pojemnik* Pojemnik::Ojciec ( ) [inline]

Definition at line 25 of file Pojemnik.hh.

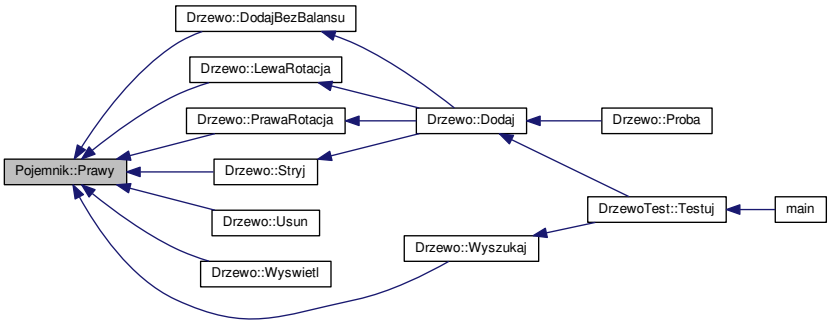
Here is the caller graph for this function:



4.6.2.3 Pojemnik* Pojemnik::Prawy ( ) [inline]

Definition at line 27 of file Pojemnik.hh.

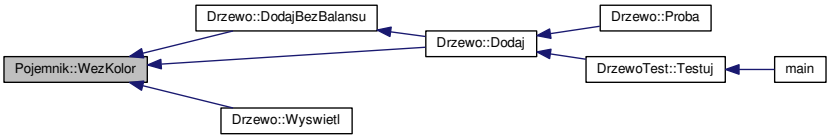
Here is the caller graph for this function:



4.6.2.4 string& Pojemnik::WezKolor ( ) [inline]

Definition at line 24 of file Pojemnik.hh.

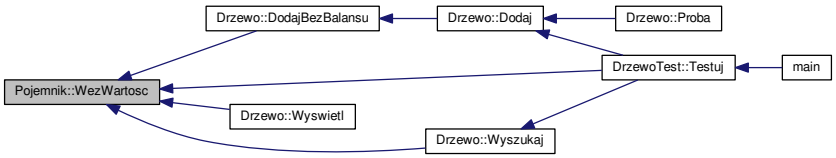
Here is the caller graph for this function:



4.6.2.5 int& Pojemnik::WezWartosc ( ) [inline]

Definition at line 23 of file Pojemnik.hh.

Here is the caller graph for this function:



4.6.3 Member Data Documentation

4.6.3.1 string Pojemnik::kolor ="R" [private]

Definition at line 16 of file Pojemnik.hh.

4.6.3.2 Pojemnik* Pojemnik::lewy =NULL

Definition at line 20 of file Pojemnik.hh.

4.6.3.3 Pojemnik* Pojemnik::ojciec =NULL

Definition at line 19 of file Pojemnik.hh.

4.6.3.4 Pojemnik* Pojemnik::prawy =NULL

Definition at line 21 of file Pojemnik.hh.

4.6.3.5 int Pojemnik::wartosc =0 [private]

Definition at line 15 of file Pojemnik.hh.

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

- Pojemnik.hh

4.7 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

4.7.1 Detailed Description

template<typename typ>class PojemnikWide< typ >

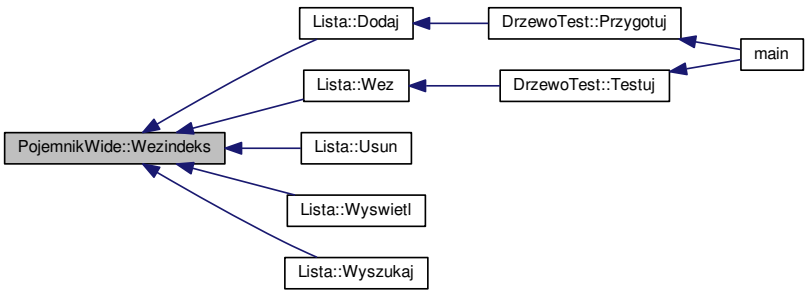
Definition at line 13 of file PojemnikWide.hh.

4.7.2 Member Function Documentation

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

Definition at line 20 of file PojemnikWide.hh.

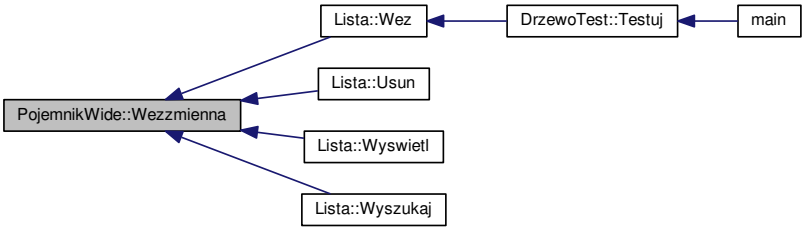
Here is the caller graph for this function:



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

Definition at line 21 of file PojemnikWide.hh.

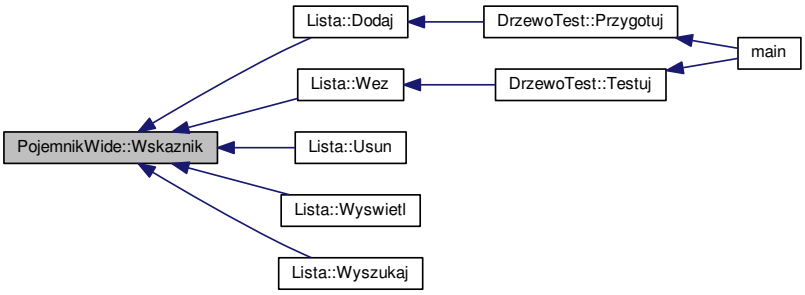
Here is the caller graph for this function:



4.7.2.3 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:



4.7.2.4 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:

4.7.3 Member Data Documentation

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

Definition at line 15 of file PojemnikWide.hh.

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

Definition at line 17 of file PojemnikWide.hh.

4.7.3.3 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

4.8 Stoper Class Reference

`#include <Stoper.hh>`

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

4.8.1 Detailed Description

Definition at line 11 of file Stoper.hh.

4.8.2 Member Function Documentation

4.8.2.1 double Stoper::get_time ( )

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

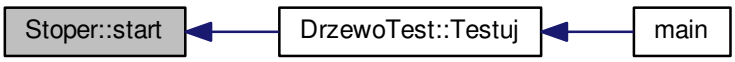
Definition at line 9 of file Stoper.cpp.

Here is the caller graph for this function:

4.8.2.2 void Stoper::start ( ) [inline]

Definition at line 16 of file Stoper.hh.

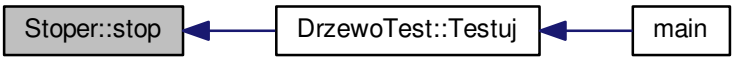
Here is the caller graph for this function:



4.8.2.3 void Stoper::stop ( ) [inline]

Definition at line 17 of file Stoper.hh.

Here is the caller graph for this function:

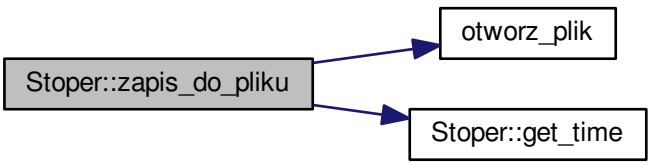


4.8.2.4 bool Stoper::zapis_do_pliku ( )

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

Definition at line 43 of file Stoper.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

4.8.3 Member Data Documentation

4.8.3.1 timeval Stoper::czas1 [private]

Definition at line 12 of file Stoper.hh.

4.8.3.2 timeval Stoper::czas2 [private]

Definition at line 13 of file Stoper.hh.

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

- Stoper.hh
- Stoper.cpp

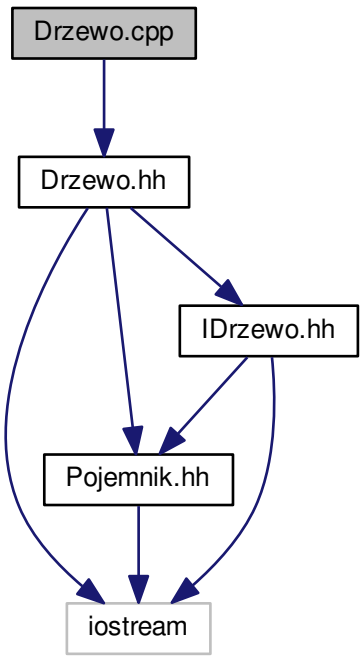
Chapter 5

File Documentation

5.1 Drzewo.cpp File Reference

```
#include "Drzewo.hh"
```

Include dependency graph for Drzewo.cpp:

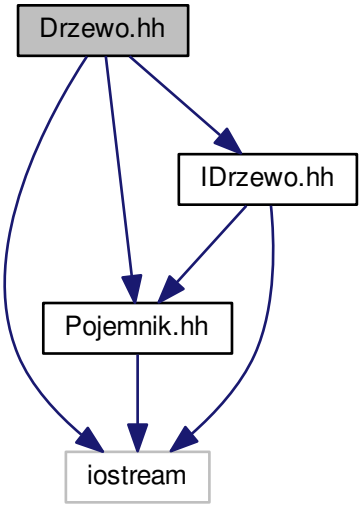


5.2 Drzewo.hh File Reference

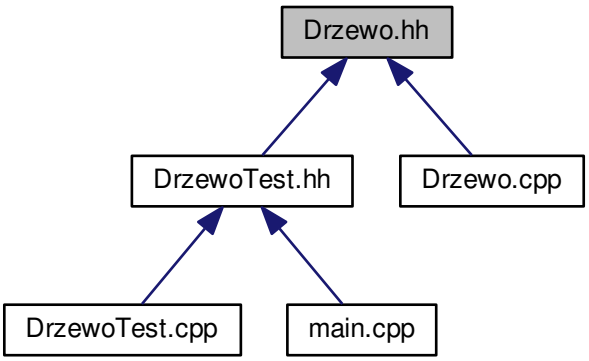
Implementacja drzewa czerwono-czarnego.

```
#include <iostream>
#include "Pojemnik.hh"
#include "IDrzewo.hh"
```

Include dependency graph for Drzewo.hh:



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



Classes

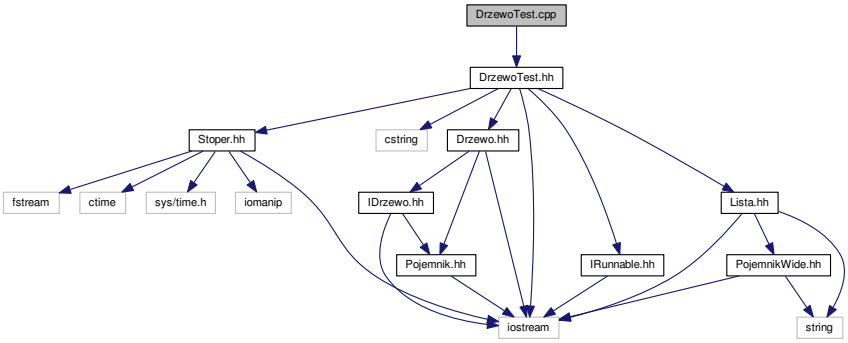
- class Drzewo

5.3 DrzewoTest.cpp File Reference

Definicja metod zwiazanych z "TablicaAsocTest".

```
#include "DrzewoTest.hh"
```

Include dependency graph for DrzewoTest.cpp:



Functions

- bool otworz_plik_odczyt (string nazwapom, fstream &StrmPlikowy)

  otwarcie pliku

5.3.1 Function Documentation

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

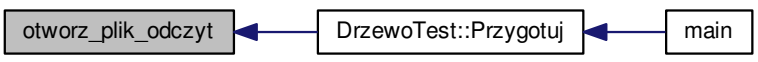
Otwiera plik i tworzy strumien do odczytu

Parameters

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

Definition at line 17 of file DrzewoTest.cpp.

Here is the caller graph for this function:

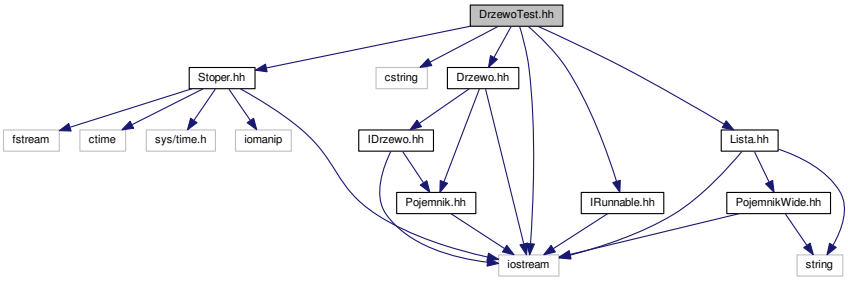


5.4 DrzewoTest.hh File Reference

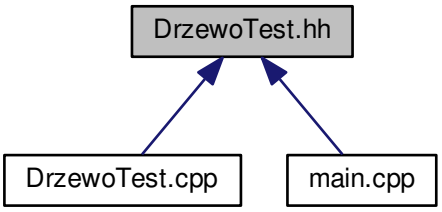
Test dzialania drzewa czerwono-czarnego.

```
#include <iostream>
#include <cstring>
#include "Stoper.hh"
#include "Drzewo.hh"
#include "Lista.hh"
#include "IRunnable.hh"
```

Include dependency graph for DrzewoTest.hh:



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



Classes

- class DrzewoTest

Functions

- bool otworz_plik_odczyt (string nazwapom, fstream &StrmPlikowy)

  otwarcie pliku

5.4.1 Function Documentation

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

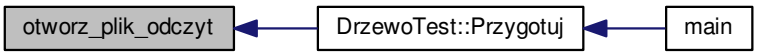
Otwiera plik i tworzy strumien do odczytu

Parameters

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

Definition at line 17 of file DrzewoTest.cpp.

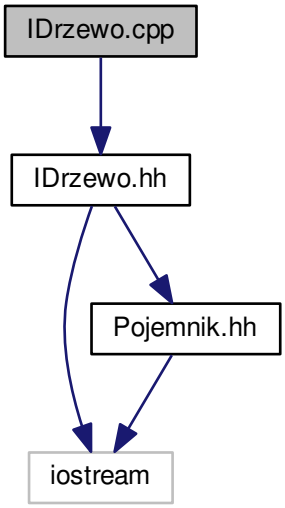
Here is the caller graph for this function:



5.5 IDrzewo.cpp File Reference

```
#include "IDrzewo.hh"
```

Include dependency graph for IDrzewo.cpp:

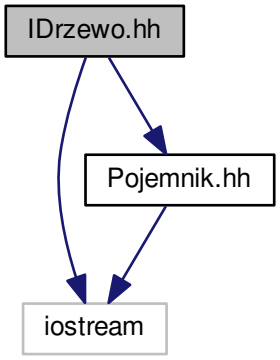


5.6 IDrzewo.hh File Reference

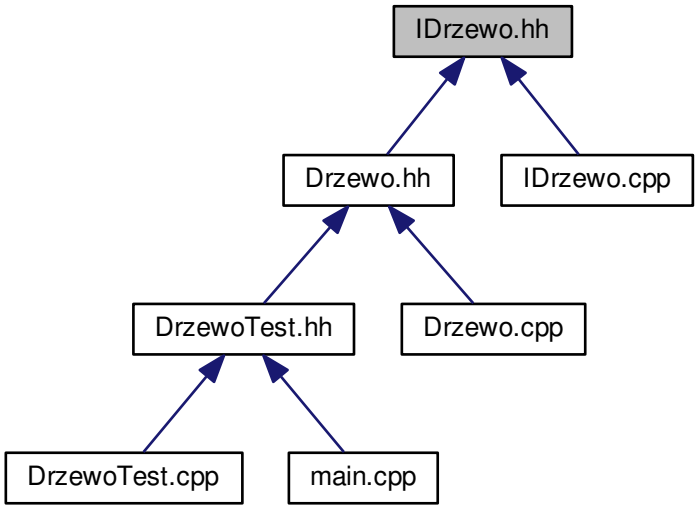
Interface drzewa czerwono-czarnego.

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

Include dependency graph for IDrzewo.hh:



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



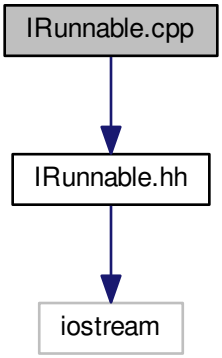
Classes

- class IDrzewo

5.7 IRunnable.cpp File Reference

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

Include dependency graph for IRunnable.cpp:

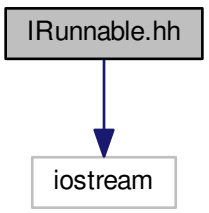


5.8 IRunnable.hh File Reference

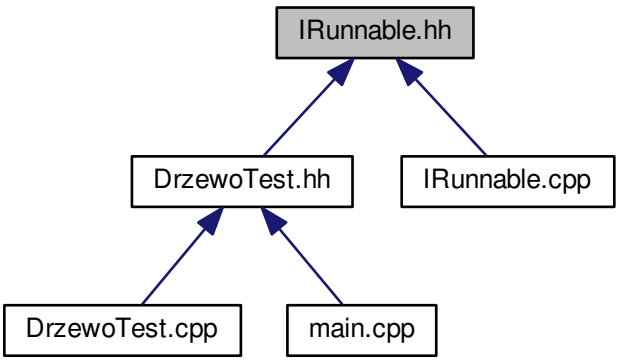
Interface testowania Drzewa czerwono-czarnego.

```
#include <iostream>
```

Include dependency graph for IRunnable.hh:



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



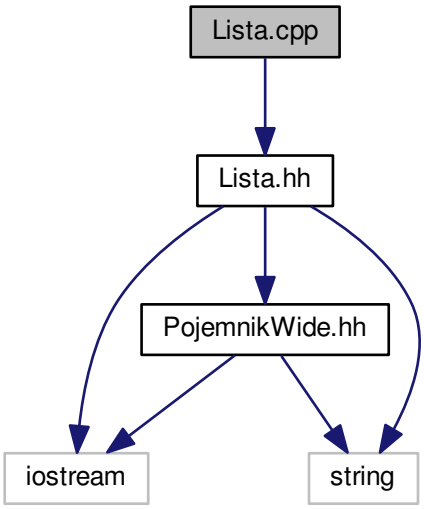
Classes

- class IRunnable

5.9 Lista.cpp File Reference

```
#include "Lista.hh"
```

Include dependency graph for Lista.cpp:

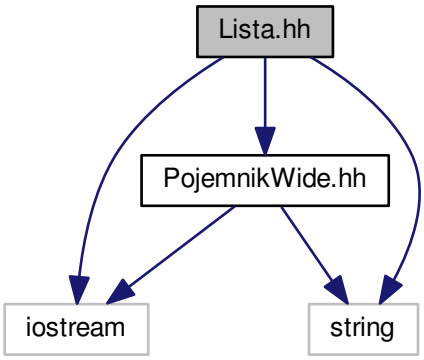


5.10 Lista.hh File Reference

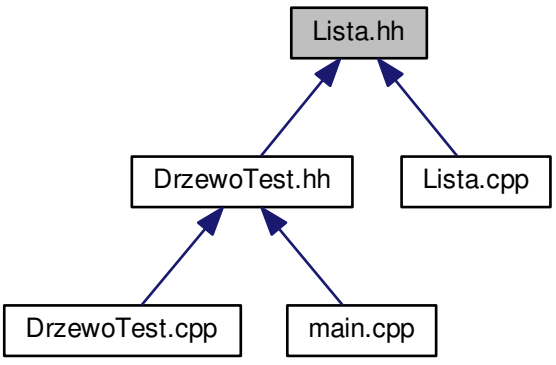
interface abstrakcyjnego typu danych - Lista

```
#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 >

5.10.1 Detailed Description

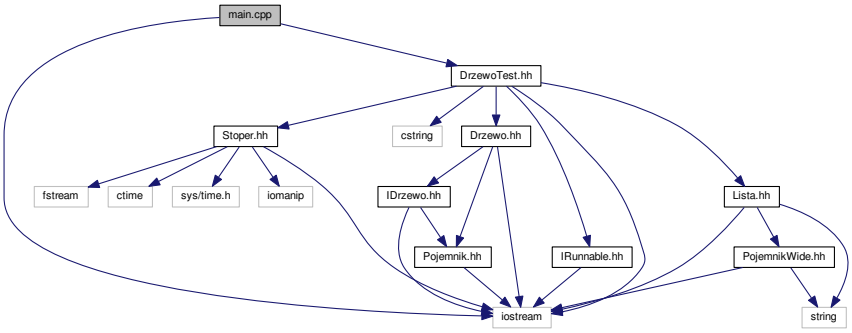
Elementy do Listy mozesz dodawac lub usuwac dowolnie czyli na poczatku, koncu badz wewnatrz listy

Definition in file Lista.hh.

5.11 main.cpp File Reference

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

Include dependency graph for main.cpp:



Functions

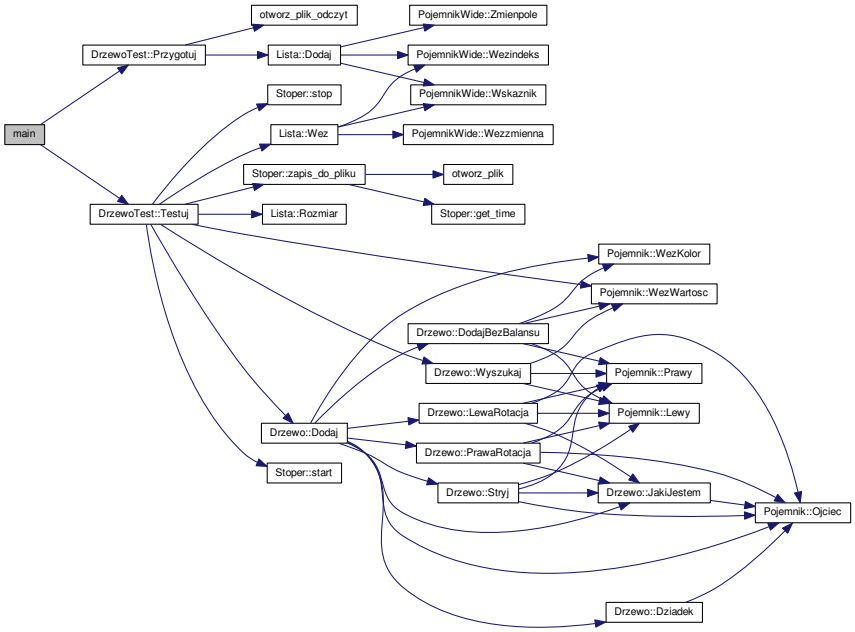
- int main ()

5.11.1 Function Documentation

5.11.1.1 int main ( )

Definition at line 7 of file main.cpp.

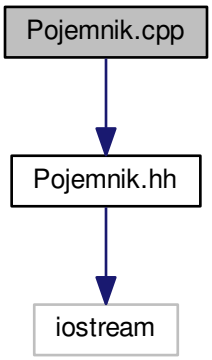
Here is the call graph for this function:



5.12 Pojemnik.cpp File Reference

```
#include "Pojemnik.hh"
```

Include dependency graph for Pojemnik.cpp:

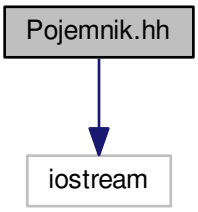


5.13 Pojemnik.hh File Reference

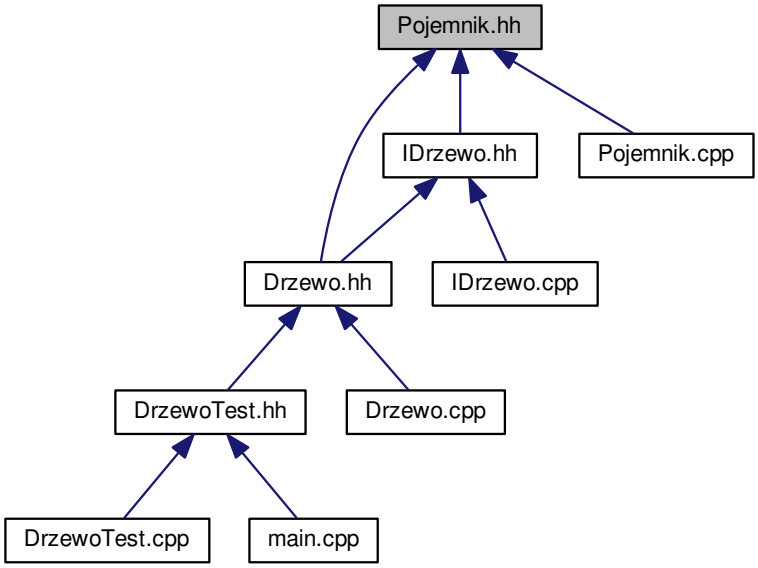
Implementacja pojedynczego elementu drzewa.

```
#include <iostream>
```

Include dependency graph for Pojemnik.hh:



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



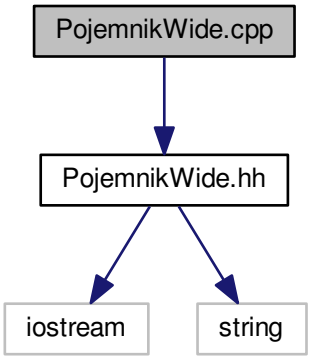
Classes

- class Pojemnik

5.14 PojemnikWide.cpp File Reference

Definicje metod pojedynczego elementu ADT (Lista)

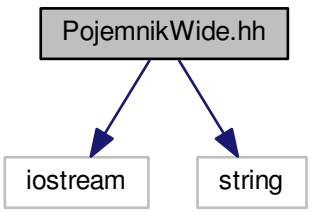
```
#include "PojemnikWide.hh"
```
Include dependency graph for PojemnikWide.cpp:



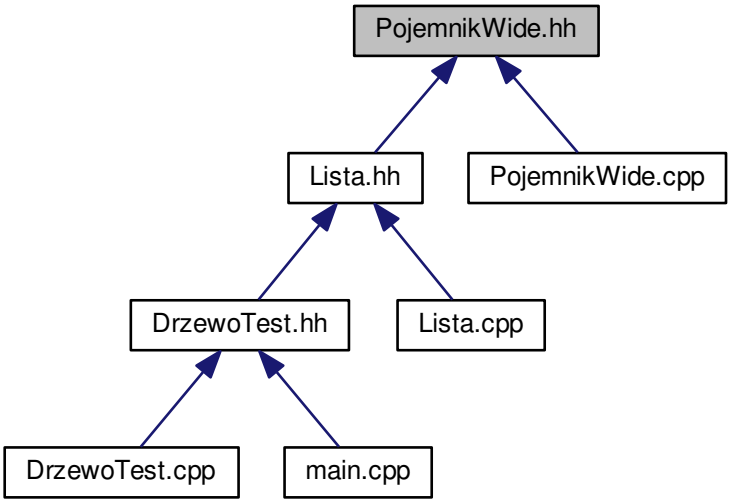
5.15 PojemnikWide.hh File Reference

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:

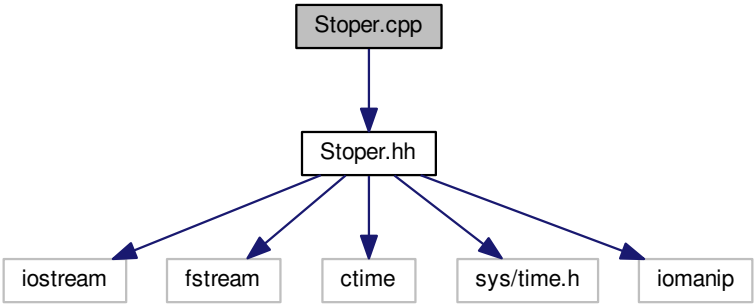


Classes

- class PojemnikWide< typ >

5.16 Stoper.cpp File Reference

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



Functions

- bool otworz_plik (string nazwapom, ofstream &StrmPlikowy)

otwarcie pliku

5.16.1 Function Documentation

5.16.1.1 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

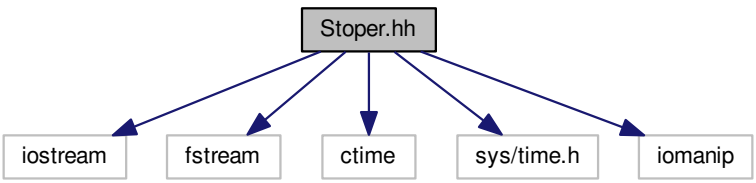
Definition at line 23 of file Stoper.cpp.

Here is the caller graph for this function:

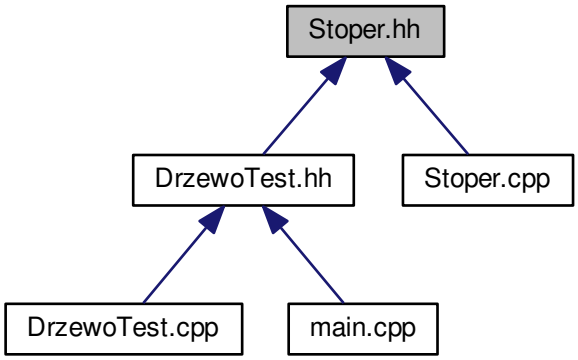
5.17 Stoper.hh File Reference

```
#include <iostream>
#include <fstream>
#include <ctime>
#include <sys/time.h>
#include <iomanip>
```

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

5.17.1 Function Documentation

5.17.1.1 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: