ADT

0.1

Generated by Doxygen 1.8.6

Sun Apr 3 2016 23:24:53

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:

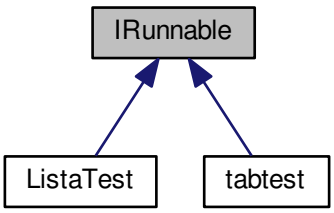
Chapter 4

Class Documentation

4.1 IRunnable Class Reference

`#include <IRunnable.hh>`

Inheritance diagram for IRunnable:



Public Member Functions

- virtual void Przygotuj (int pilosc, int ppowtorzenia, char popcja)=0
- virtual void Przygotuj (string pnazwapliku, string pszukane)=0
- virtual int Testuj ()=0

4.1.1 Detailed Description

Definition at line 12 of file IRunnable.hh.

4.1.2 Member Function Documentation

4.1.2.1 virtual void IRunnable::Przygotuj (int *pilosc,* int *ppowtorzenia,* char *popcja*) [pure virtual]

Implemented in ListaTest, and tabtest.

4.1.2.2 virtual void IRunnable::Przygotuj (string *pnazwapliku,* string *pszukane*) [pure virtual]

Implemented in ListaTest, and tabtest.

4.1.2.3 virtual int IRunnable::Testuj ( ) [pure virtual]

Implemented in ListaTest, and tabtest.

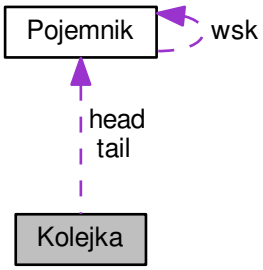
The documentation for this class was generated from the following file:

- IRunnable.hh

4.2 Kolejka Class Reference

`#include <Kolejka.hh>`

Collaboration diagram for Kolejka:



Public Member Functions

- void Dodaj (double elem)
- double Zwroclusun ()
- double Wez ()
- int Rozmiar ()
- bool Czypusta ()
- void Oproznij ()
- void Wyswietl ()

Private Attributes

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

4.2.1 Detailed Description

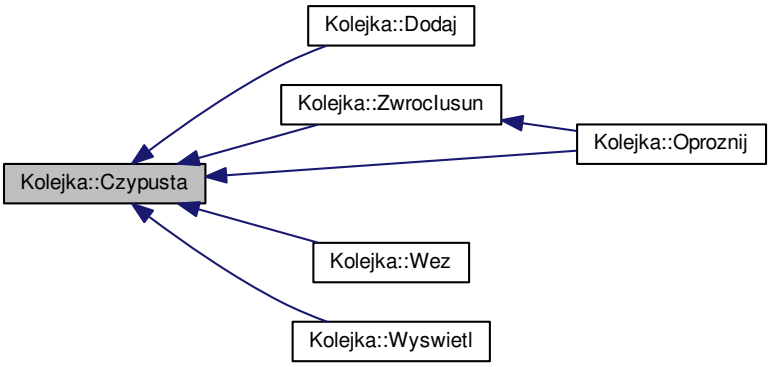
Definition at line 13 of file Kolejka.hh.

4.2.2 Member Function Documentation

4.2.2.1 bool Kolejka::Czypusta ( ) [inline]

Definition at line 22 of file Kolejka.hh.

Here is the caller graph for this function:



4.2.2.2 void Kolejka::Dodaj (double *elem*)

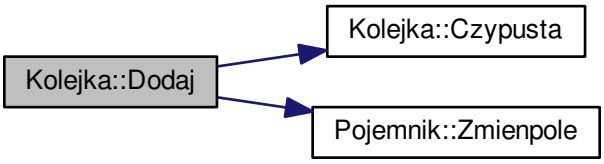
Dodaje element na koncu kolejki

Parameters

in	*elem*-	zmienna do przechowania

Definition at line 13 of file Kolejka.cpp.

Here is the call graph for this function:

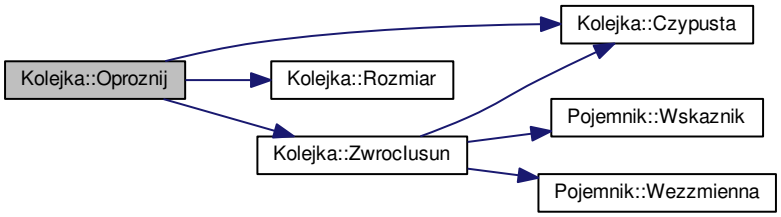


4.2.2.3 void Kolejka::Oproznij ( )

Usuwa wszystkie elementy kolejki

Definition at line 73 of file Kolejka.cpp.

Here is the call graph for this function:



4.2.2.4 int Kolejka::Rozmiar ( ) [inline]

Definition at line 21 of file Kolejka.hh.

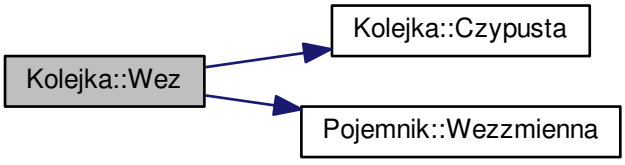
Here is the caller graph for this function:

4.2.2.5 double 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:

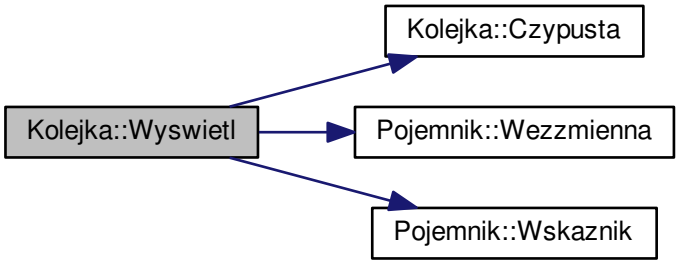


4.2.2.6 void Kolejka::Wyswietl ( )

Wyswietla wszystkie elementy kolejki od pierwszego do ostatniego

Definition at line 84 of file Kolejka.cpp.

Here is the call graph for this function:



4.2.2.7 double Kolejka::Zwroclusun ( )

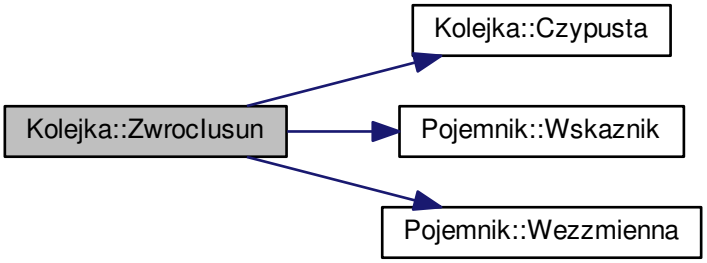
Usuwa element z poczatku kolejki i zwraca jego wartosc

Return values

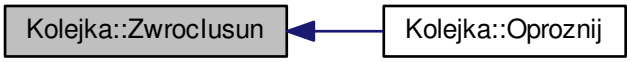
wartosc	usunietego elementu

Definition at line 37 of file Kolejka.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



4.2.3 Member Data Documentation

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

Definition at line 14 of file Kolejka.hh.

4.2.3.2 int Kolejka::rozmiar =0 [private]

Definition at line 16 of file Kolejka.hh.

4.2.3.3 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

4.3 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

4.3.1 Detailed Description

template<typename typ>class Lista< typ >

Definition at line 18 of file Lista.hh.

4.3.2 Member Function Documentation

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

Return values

true-	gdy lista jest pusta
false-	w przypadku przeciwnym

Definition at line 31 of file Lista.hh.

4.3.2.2 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

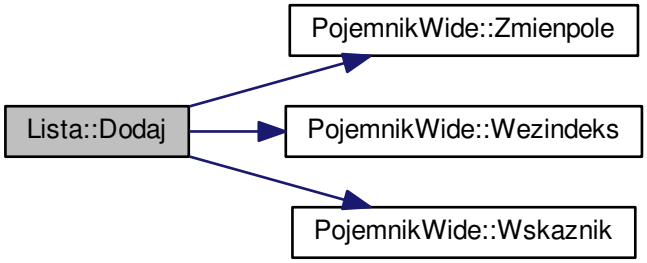
in	*elem-*	wartosc do przechowania
in	*index-*	indeks listy pod jakim bedzie przechowywany pojemnik ze zmienna

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 52 of file Lista.hh.

Here is the call graph for this function:



Here is the caller graph for this function:

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

Usuwa wszystkie elementy z listy

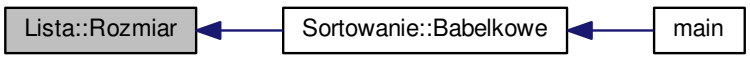
Definition at line 227 of file Lista.hh.

Here is the caller graph for this function:

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

Definition at line 26 of file Lista.hh.

Here is the caller graph for this function:



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

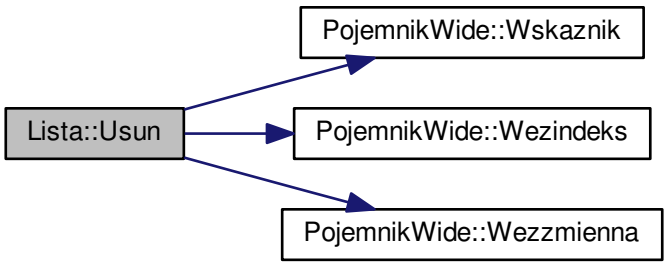
Usuwa element z Listy o zadanym indeksie i zwraca wartosc, ktora przechowywal

Parameters

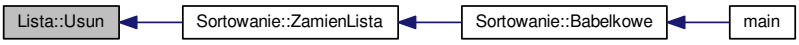
in	*ind-*	indeks elementu, ktory ma zostac usuniety z listy

Definition at line 145 of file Lista.hh.

Here is the call graph for this function:



Here is the caller graph for this function:



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

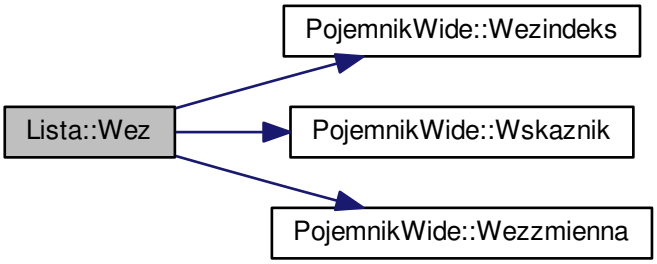
Zwraca wartość elementu o zadanym indeksie

Parameters

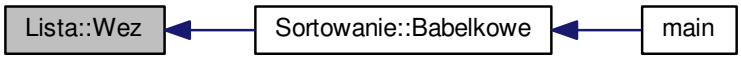
in	*ind*-	indeks poszukiwanego elementu

Definition at line 117 of file Lista.hh.

Here is the call graph for this function:



Here is the caller graph for this function:

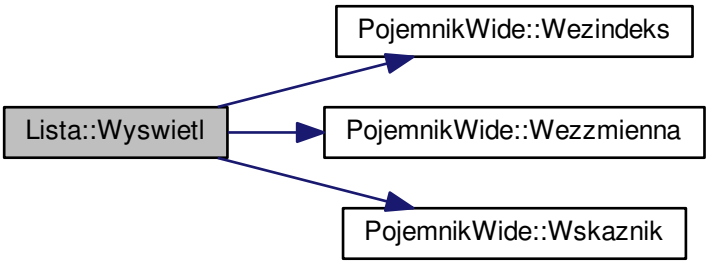


4.3.2.7 template<typename typ > void Lista< typ >::Wyswietl ( )

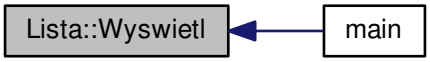
Wyswietla zawartosc listy na standardowe wyjscie

Definition at line 241 of file Lista.hh.

Here is the call graph for this function:



Here is the caller graph for this function:



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

Wyszukuje podany wyraz wsrod elementow listy

Parameters

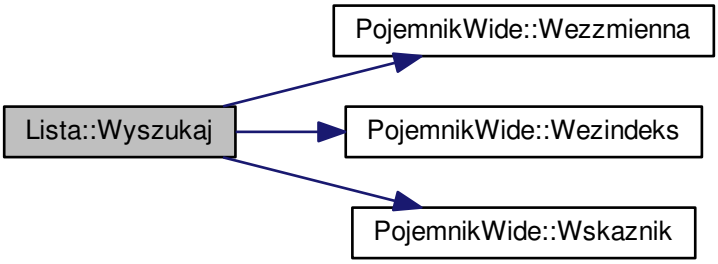
in	*szukane-*	szukany wyraz

Return values

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

Definition at line 268 of file Lista.hh.

Here is the call graph for this function:



4.3.3 Member Data Documentation

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

Definition at line 19 of file Lista.hh.

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

Definition at line 20 of file Lista.hh.

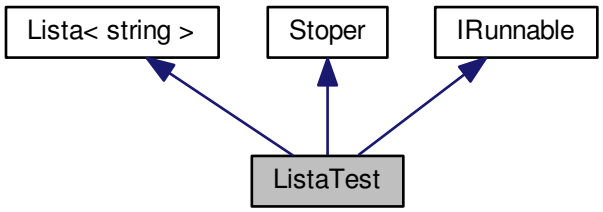
The documentation for this class was generated from the following file:

- Lista.hh

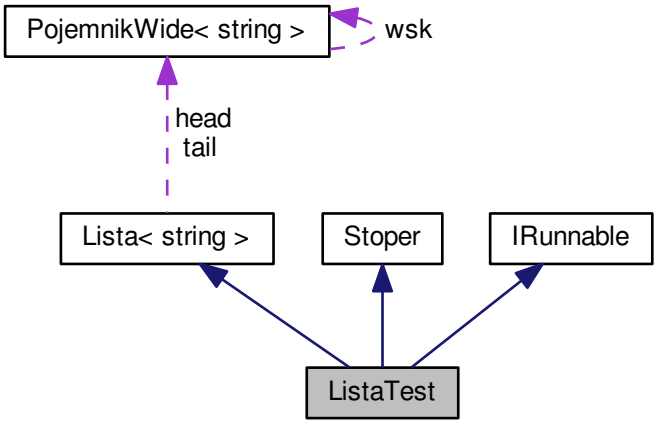
4.4 ListaTest Class Reference

`#include <ListaTest.hh>`

Inheritance diagram for ListaTest:



Collaboration diagram for ListaTest:



Public Member Functions

- void Przygotuj (string pnazwapliku, string pszukane)

  przygotowanie do testu przeszukiwania listy
- int Testuj ()
- void Przygotuj (int pilosc, int ppowtorzenia, char popcja)
- bool OdczytajIzapisz (string nazwapom)

Private Attributes

- string nazwapliku
- string szukane

4.4.1 Detailed Description

Definition at line 19 of file ListaTest.hh.

4.4.2 Member Function Documentation

4.4.2.1 bool ListaTest::OdczytajIzapisz (string *nazwapom*)

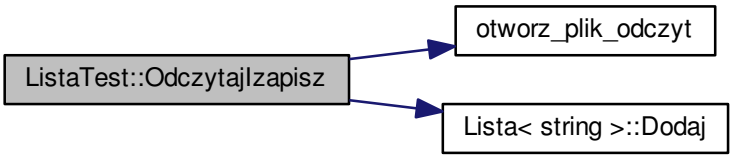
Funkcja odpowiedzialna za odczytanie danych z pliku i zapisanie ich w liscie

Parameters

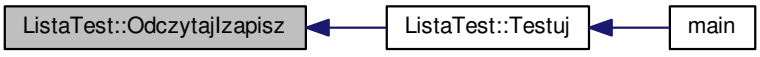
in	*nazwapom-*	nazwa pliku, skad mamy odczytywac tekst

Definition at line 36 of file ListaTest.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



4.4.2.2 void ListaTest::Przygotuj (string *pnazwapliku,* string *pszukane*) [virtual]

Parameters

in	*pnazwapliku-*	nazwa pliku, zktorego ma byc wczytany tekst i zapisany do listy

in	*szukane-*	szukane slowo

Implements IRunnable.

Definition at line 59 of file ListaTest.cpp.

Here is the caller graph for this function:

4.4.2.3 void ListaTest::Przygotuj (int *pilosc,* int *ppowtorzenia,* char *popcja*) [inline], [virtual]

Implements IRunnable.

Definition at line 26 of file ListaTest.hh.

4.4.2.4 int ListaTest::Testuj ( ) [virtual]

Wczytuje dane z pliku, zapisuje do listy, mierzy czas wyszukiwania elementu i dopisuje go do pliku "czasy.dat"

Parameters

in	*nazwapliku-*	nazwa pliku z tekstem do zapisania
in	*szukane-*	szukane slowo

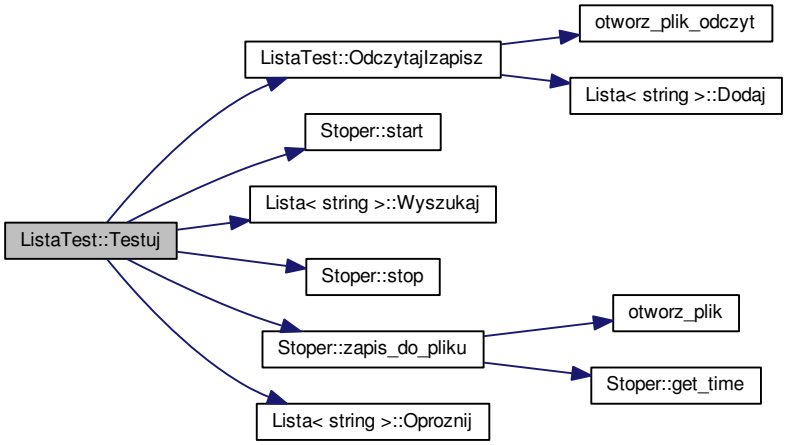
Return values

numer	indeksu pod jakim znajdziemy szukane slowo w liscie lub -1 gdy slowo nie wystepuje w tekscie

Implements IRunnable.

Definition at line 74 of file ListaTest.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

4.4.3 Member Data Documentation

4.4.3.1 string ListaTest::nazwapliku [private]

Definition at line 20 of file ListaTest.hh.

4.4.3.2 string ListaTest::szukane [private]

Definition at line 21 of file ListaTest.hh.

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

- ListaTest.hh
- ListaTest.cpp

4.5 Pojemnik Class Reference

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

Collaboration diagram for Pojemnik:

Public Member Functions

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

Public Attributes

- Pojemnik ∗ wsk =NULL

Private Attributes

- double zmienna =0

4.5.1 Detailed Description

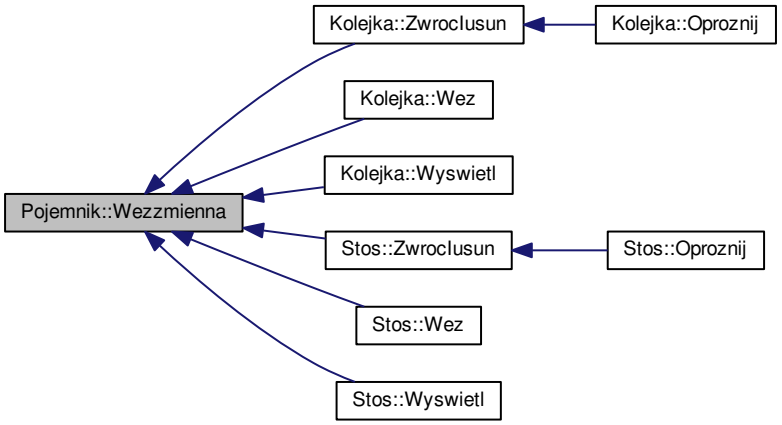
Definition at line 12 of file Pojemnik.hh.

4.5.2 Member Function Documentation

4.5.2.1 double Pojemnik::Wezzmienna ( ) [inline]

Definition at line 18 of file Pojemnik.hh.

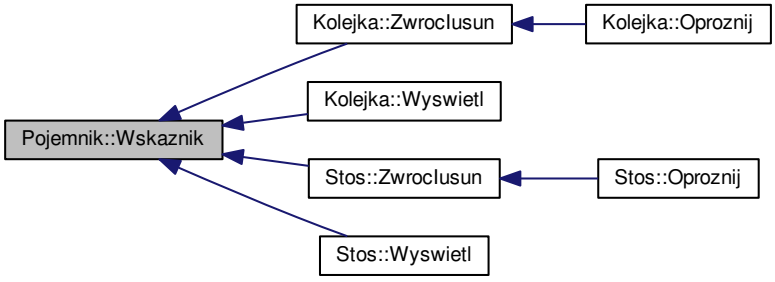
Here is the caller graph for this function:



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

Definition at line 19 of file Pojemnik.hh.

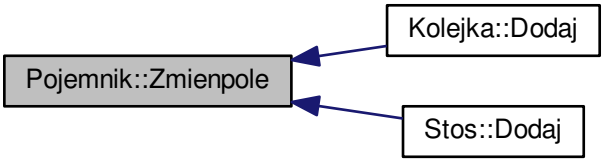
Here is the caller graph for this function:



4.5.2.3 void Pojemnik::Zmienpole (double *pom*) [inline]

Definition at line 17 of file Pojemnik.hh.

Here is the caller graph for this function:



4.5.3 Member Data Documentation

4.5.3.1 Pojemnik* Pojemnik::wsk =NULL

Definition at line 15 of file Pojemnik.hh.

4.5.3.2 double Pojemnik::zmienna =0 [private]

Definition at line 13 of file Pojemnik.hh.

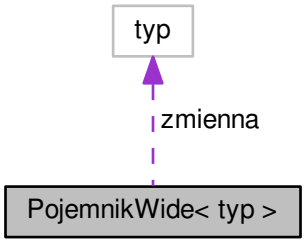
The documentation for this class was generated from the following file:

- Pojemnik.hh

4.6 PojemnikWide< typ > Class Template Reference

`#include <PojemnikWide.hh>`

Collaboration diagram for PojemnikWide< typ >:



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

4.6.1 Detailed Description

template<typename typ>class PojemnikWide< typ >

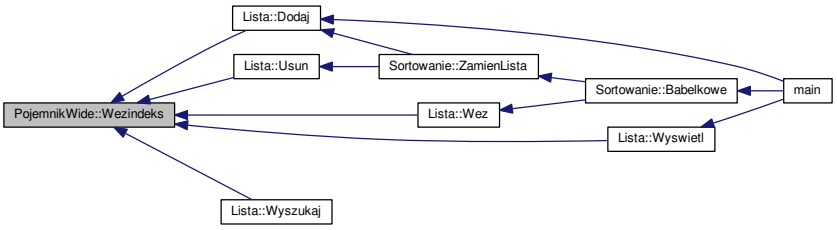
Definition at line 13 of file PojemnikWide.hh.

4.6.2 Member Function Documentation

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

Definition at line 20 of file PojemnikWide.hh.

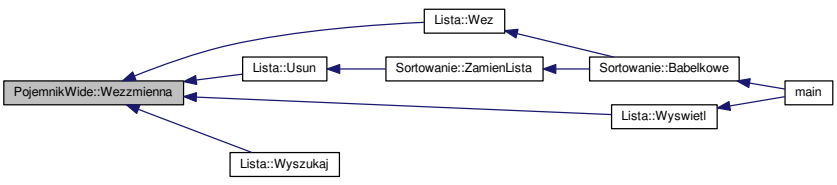
Here is the caller graph for this function:



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

Definition at line 21 of file PojemnikWide.hh.

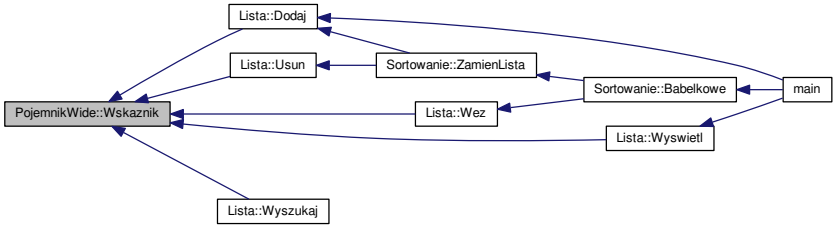
Here is the caller graph for this function:



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

Definition at line 22 of file PojemnikWide.hh.

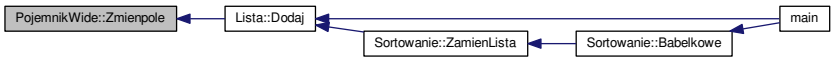
Here is the caller graph for this function:



4.6.2.4 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:



4.6.3 Member Data Documentation

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

Definition at line 15 of file PojemnikWide.hh.

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

Definition at line 17 of file PojemnikWide.hh.

4.6.3.3 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

4.7 Sortowanie Class Reference

`#include <Sortowanie.hh>`

Public Member Functions

- void ZamienLista (Lista< double > &lista, int i, int j)
- void Babelkowe (Lista< double > &lista)

  Sortowanie babelkowe.

4.7.1 Detailed Description

Definition at line 15 of file Sortowanie.hh.

4.7.2 Member Function Documentation

4.7.2.1 void Sortowanie::Babelkowe (Lista< double > & *lista*)

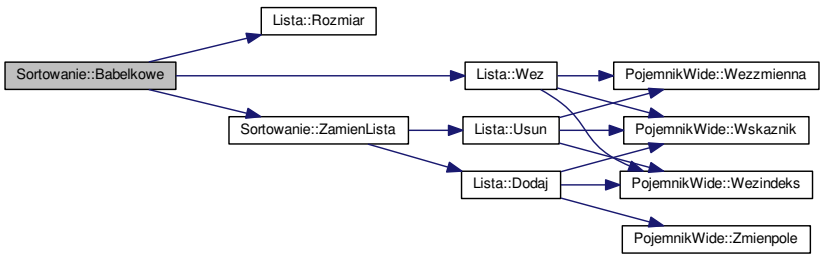
Metoda do sortowania elementow w liscie

Parameters

in	*lista-lista,ktora*	mamy posortowac

Definition at line 35 of file Sortowanie.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

4.7.2.2 void Sortowanie::ZamienLista (Lista< double > & *lista*, int *i*, int *j*)

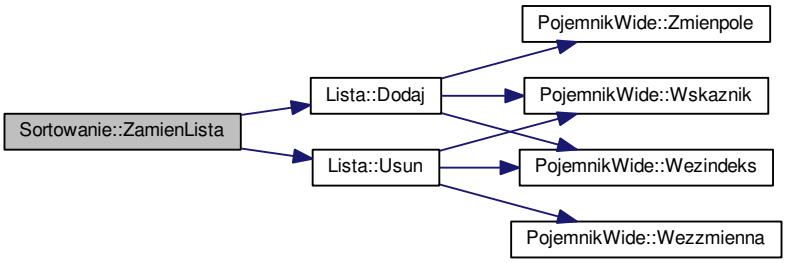
Dokonuje zamiany elementow w Liscie

Parameters

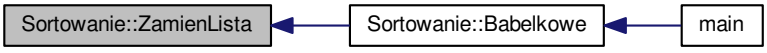
in	*lista-*	obiekt poddany dzialaniu
in	*i-*	numer indeksu elementu listy
in	*j-*	numer indeksu elementu listy

Definition at line 15 of file Sortowanie.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



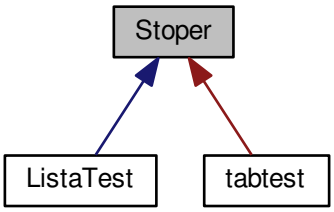
The documentation for this class was generated from the following files:

- Sortowanie.hh
- Sortowanie.cpp

4.8 Stoper Class Reference

`#include <Stoper.hh>`

Inheritance 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

4.8.1 Detailed Description

Definition at line 11 of file Stoper.hh.

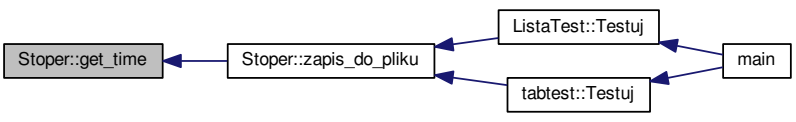
4.8.2 Member Function Documentation

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.

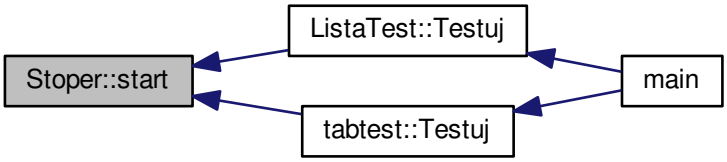
Here is the caller graph for this function:



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.

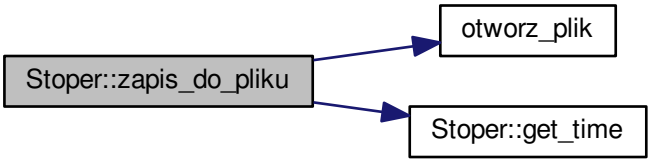
Here is the caller graph for this function:

4.8.2.4 bool Stoper::zapis_do_pliku ( )

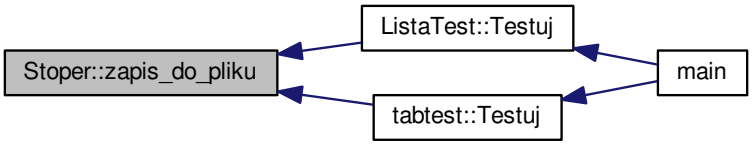
Wywolanie tej funkcji skutkuje dopisaniem do pliku "czasy.dat" ostatniej roznicy czasowej ("czas_stop"-"czas_start) Wartosci wyrazone w sekundach

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.

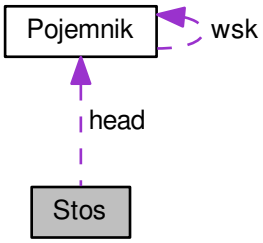
The documentation for this class was generated from the following files:

- Stoper.hh
- Stoper.cpp

4.9 Stos Class Reference

`#include <Stos.hh>`

Collaboration diagram for Stos:



Public Member Functions

- void Dodaj (double elem)
- double Zwroclusun ()
- double Wez ()
- bool Czypusty ()
- int Rozmiar ()
- void Oproznij ()
- void Wyswietl ()

Private Attributes

- Pojemnik * head =NULL
- int rozmiar =0

4.9.1 Detailed Description

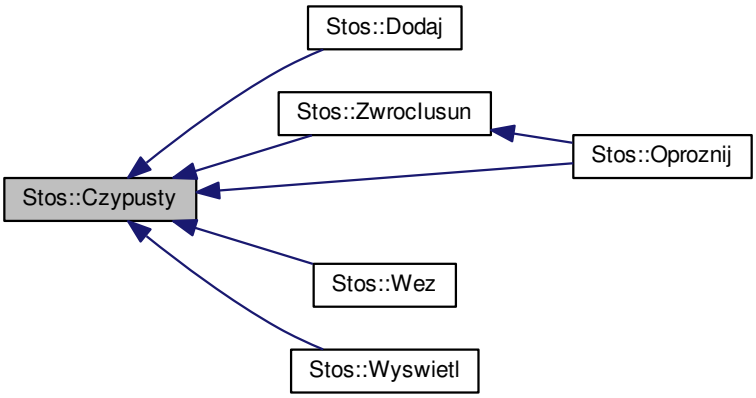
Definition at line 13 of file Stos.hh.

4.9.2 Member Function Documentation

4.9.2.1 bool Stos::Czypusty ( ) [inline]

Definition at line 20 of file Stos.hh.

Here is the caller graph for this function:



4.9.2.2 void Stos::Dodaj (double *elem*)

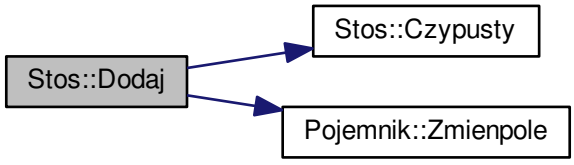
"Kladzie" element na Stos

Parameters

in	*elem*-	zmienna do przechowania

Definition at line 13 of file Stos.cpp.

Here is the call graph for this function:

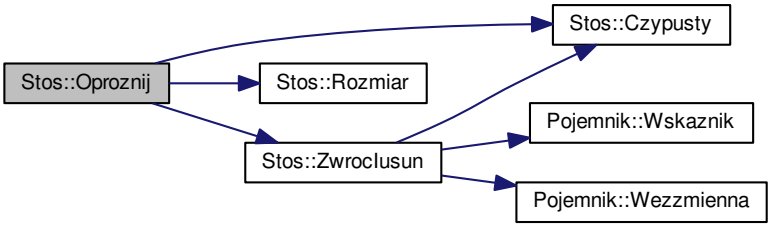


4.9.2.3 void Stos::Oproznij ( )

Usuwa wszystkie elementy stosu

Definition at line 71 of file Stos.cpp.

Here is the call graph for this function:



4.9.2.4 int Stos::Rozmiar ( ) [inline]

Definition at line 21 of file Stos.hh.

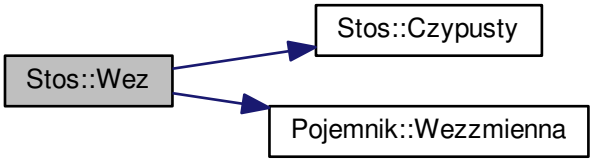
Here is the caller graph for this function:

4.9.2.5 double 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:

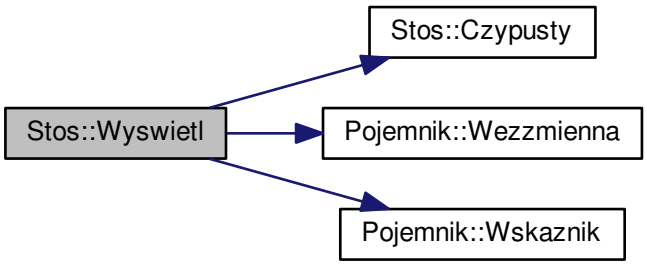


4.9.2.6 void Stos::Wyswietl ( )

Wyswietla wszystkie elementy stosu od "wierzcholka" do dolu

Definition at line 82 of file Stos.cpp.

Here is the call graph for this function:



4.9.2.7 double Stos::Zwroclusun ( )

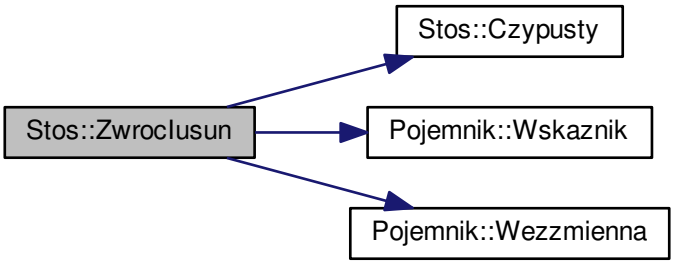
Usuwa element ze stosu

Return values

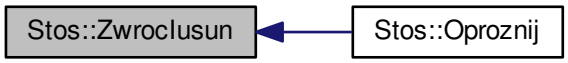
wartosc	usunietego elementu

Definition at line 35 of file Stos.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



4.9.3 Member Data Documentation

4.9.3.1 Pojemnik* Stos::head =NULL [private]

Definition at line 14 of file Stos.hh.

4.9.3.2 int Stos::rozmiar =0 [private]

Definition at line 15 of file Stos.hh.

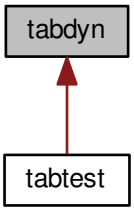
The documentation for this class was generated from the following files:

- Stos.hh
- Stos.cpp

4.10 tabdyn Class Reference

`#include <Tablica.hh>`

Inheritance diagram for tabdyn:



Public Member Functions

- void usun ()
- void wyswietl ()
- int wez_rozmiar ()
- void zainicjalizuj ()

- void dodaj_liczby (int pom)
- void dodaj_liczby_dwa (int pom)
- void dodaj_liczby_dek (int pom)
- int ile_elementow ()

Private Attributes

- int * tablica =NULL
- int licznik =0
- int rozmiar =0

4.10.1 Detailed Description

Definition at line 7 of file Tablica.hh.

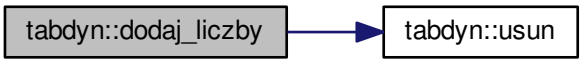
4.10.2 Member Function Documentation

4.10.2.1 void tabdyn::dodaj_liczby (int *pom*)

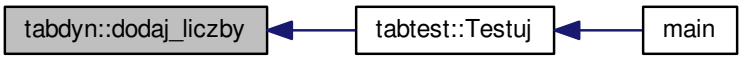
W przypadku zapelnienia tablicy dynamicznej zwieksza jej rozmiar o jeden (element typu int)

Definition at line 16 of file Tablica.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



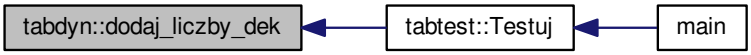
4.10.2.2 void tabdyn::dodaj_liczby_dek (int *pom*)

Gdy zabraknie miejsca w tablicy na nowy element, funkcja zwieksza ja o 10 (elementow typu int)

Definition at line 73 of file Tablica.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

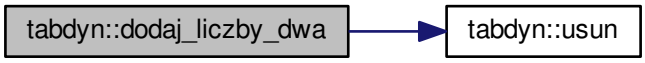


4.10.2.3 void tabdyn::dodaj_liczby_dwa (int *pom*)

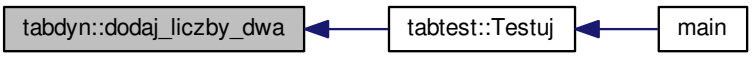
Funkcja rozni sie od "dodaj_liczby" sposobem zmieniania rozmiaru tablicy. W przypadku zapelnienia tablicy dynamicznej, funkcja alokuje nowa, dwa razy wieksza

Definition at line 44 of file Tablica.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:

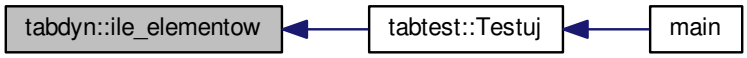


4.10.2.4 int tabdyn::ile_elementow ( ) [inline]

Zwraca ilosc elementow przechowywanych w tablicy

Definition at line 23 of file Tablica.hh.

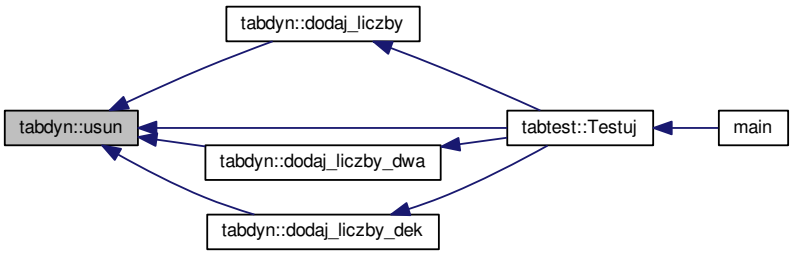
Here is the caller graph for this function:



4.10.2.5 void tabdyn::usun ( ) [inline]

Definition at line 13 of file Tablica.hh.

Here is the caller graph for this function:



4.10.2.6 int tabdyn::wez_rozmiar ( ) [inline]

Definition at line 15 of file Tablica.hh.

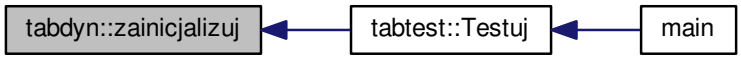
4.10.2.7 void tabdyn::wyswietl ( ) [inline]

Definition at line 14 of file Tablica.hh.

4.10.2.8 void tabdyn::zainicjalizuj ( )

Definition at line 3 of file Tablica.cpp.

Here is the caller graph for this function:



4.10.3 Member Data Documentation

4.10.3.1 int tabdyn::licznik =0 [private]

Definition at line 9 of file Tablica.hh.

4.10.3.2 int tabdyn::rozmiar =0 [private]

Definition at line 10 of file Tablica.hh.

4.10.3.3 int* tabdyn::tablica =NULL [private]

Definition at line 8 of file Tablica.hh.

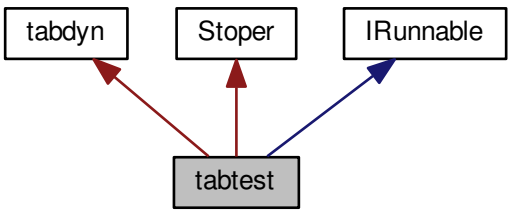
The documentation for this class was generated from the following files:

- Tablica.hh
- Tablica.cpp

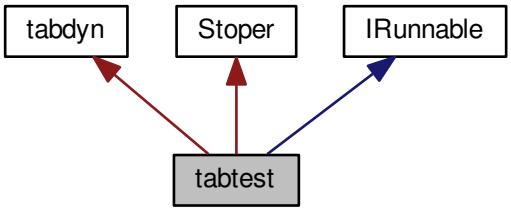
4.11 tabtest Class Reference

`#include <TablicaTest.hh>`

Inheritance diagram for tabtest:



Collaboration diagram for tabtest:



Public Member Functions

- void Przygotuj (int pilosc, int ppowtorzenia, char popcja)

  przygotowanie do testu
- int Testuj ()

  Funkcja testujaca algorytmy.
- void Przygotuj (string pnazwapliku, string pszukane)

Private Attributes

- int ilosc =10
- int powtorzenia =1
- char opcja ='1'

Additional Inherited Members

4.11.1 Detailed Description

Definition at line 10 of file TablicaTest.hh.

4.11.2 Member Function Documentation

4.11.2.1 void tabtest::Przygotuj (int *pilosc,* int *ppowtorzenia,* char *popcja*) [virtual]

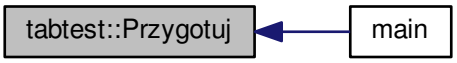
Parameters

in	*pilosc-*	ilosc lczb jaka ma byc zapisana do tablicy
in	*ppowtorzenia-*	ile razy ma byc wywolany algorytm, pomiar czasowy
in	*popcja-*	zmienna potrzebna do wyboru algorytmu (1. powiekszzanie tablicy o 1, 2. powiekszanie tablcy o 10, 3. powiekszanie tablicy 2 razy)

Implements IRunnable.

Definition at line 12 of file TablicaTest.cpp.

Here is the caller graph for this function:



4.11.2.2 void tabtest::Przygotuj (string *pnazwapliku,* string *pszukane*) [inline],[virtual]

Implements IRunnable.

Definition at line 18 of file TablicaTest.hh.

4.11.2.3 int tabtest::Testuj ( ) [virtual]

Funkcja wywoluje algorytmy dodawania do tablicy, mierzy czas ich pracy i zapisuje dane (czasy) do pliku "czasy.dat"

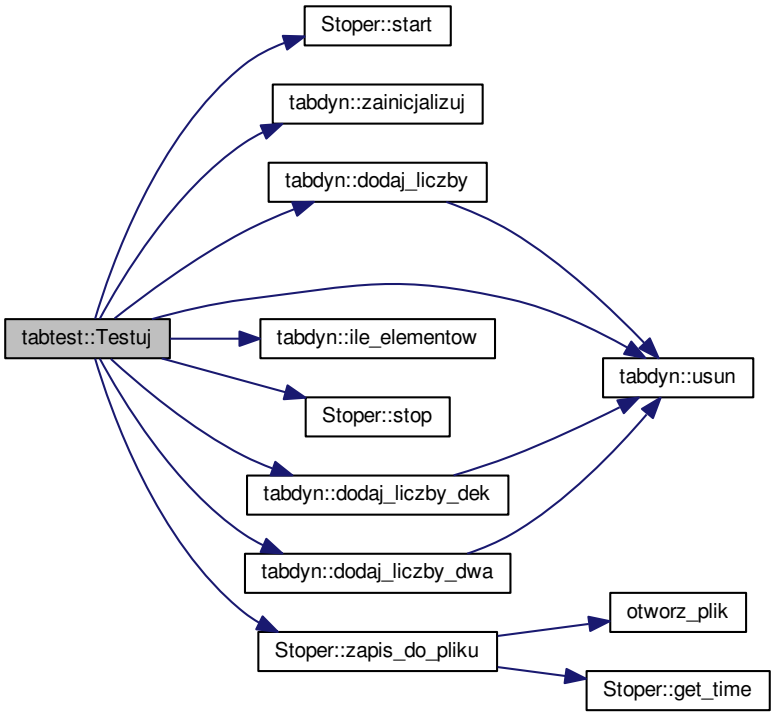
Return values

funkcja	zwraca wartosc 1 gdy wszystko wykonalo sioe poprawnie
funkcja	zwraca 0 gdy wystapil jakis blad

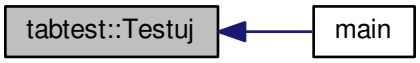
Implements IRunnable.

Definition at line 27 of file TablicaTest.cpp.

Here is the call graph for this function:



Here is the caller graph for this function:



4.11.3 Member Data Documentation

4.11.3.1 int tabtest::ilosc =10 [private]

Definition at line 11 of file TablicaTest.hh.

4.11.3.2 char tabtest::opcja ='1' [private]

Definition at line 13 of file TablicaTest.hh.

4.11.3.3 int tabtest::powtorzenia =1 [private]

Definition at line 12 of file TablicaTest.hh.

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

- TablicaTest.hh
- TablicaTest.cpp

Chapter 5

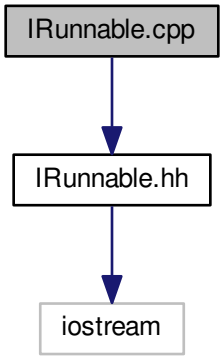
File Documentation

5.1 IRunnable.cpp File Reference

interfejs do testowania programow

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

Include dependency graph for IRunnable.cpp:

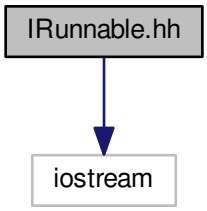


5.2 IRunnable.hh File Reference

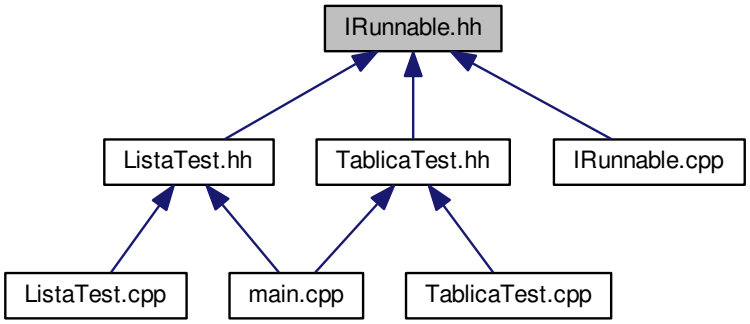
interfejs do testowania programow

```
#include <iostream>
```

Include dependency graph for IRunnable.hh:



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



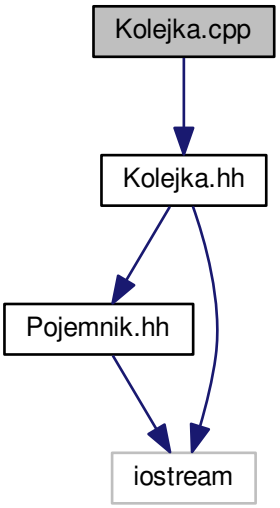
Classes

- class IRunnable

5.3 Kolejka.cpp File Reference

Definicja metod interface'u ADT- Kolejka.

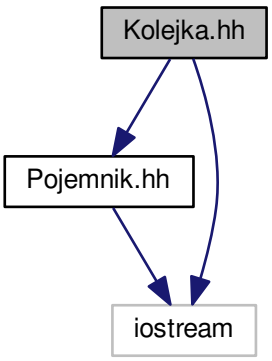
```
#include "Kolejka.hh"
```
Include dependency graph for Kolejka.cpp:



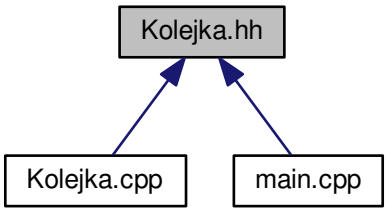
5.4 Kolejka.hh File Reference

interface 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:

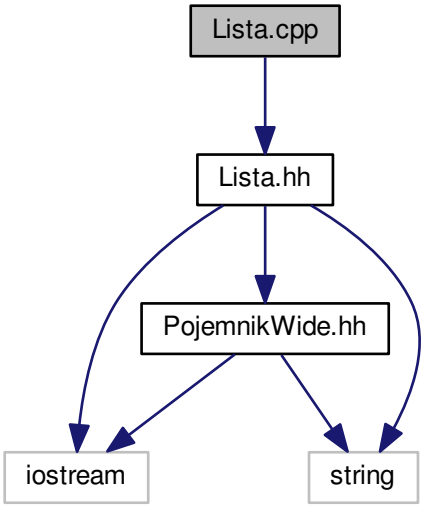


Classes

- class Kolejka

5.5 Lista.cpp File Reference

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

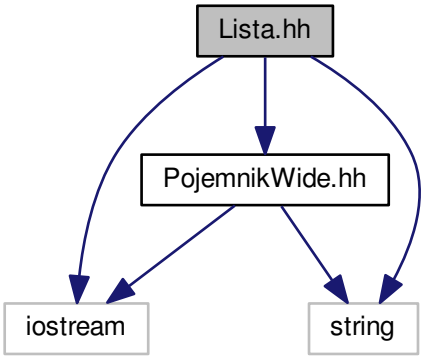


5.6 Lista.hh File Reference

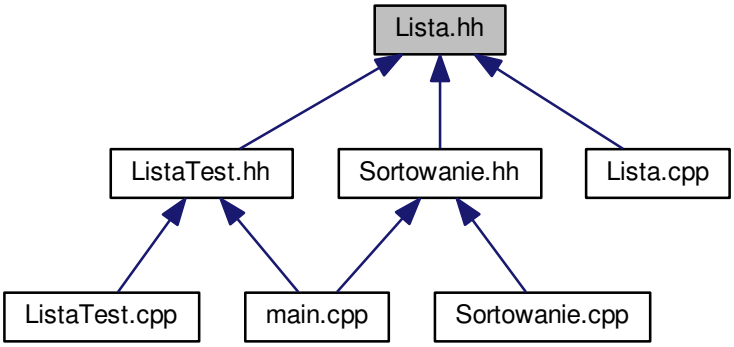
interface abstrakcyjnego typu danych - Lista

```
#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 >

5.6.1 Detailed Description

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

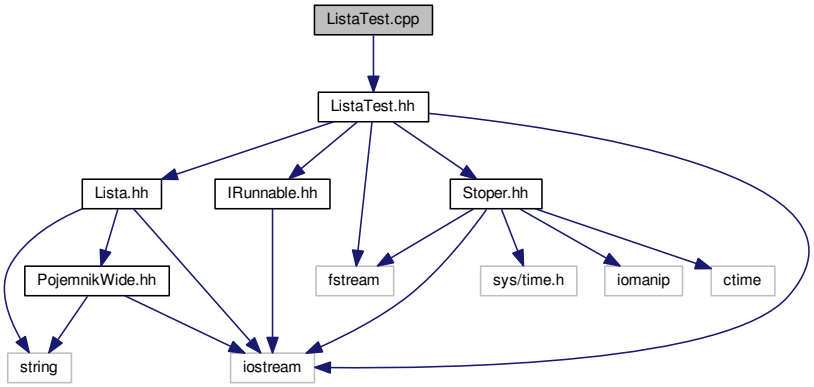
Definition in file Lista.hh.

5.7 ListaTest.cpp File Reference

Definicja metod zwiazanych z "ListaTest".

```
#include "ListaTest.hh"
```

Include dependency graph for ListaTest.cpp:



Functions

- bool otworz_plik_odczyt (string nazwapom, fstream &StrmPlikowy)

  otwarcie pliku

5.7.1 Function Documentation

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

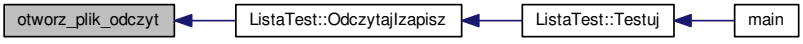
Otwiera plik i tworzy strumien do odczytu

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 ListaTest.cpp.

Here is the caller graph for this function:

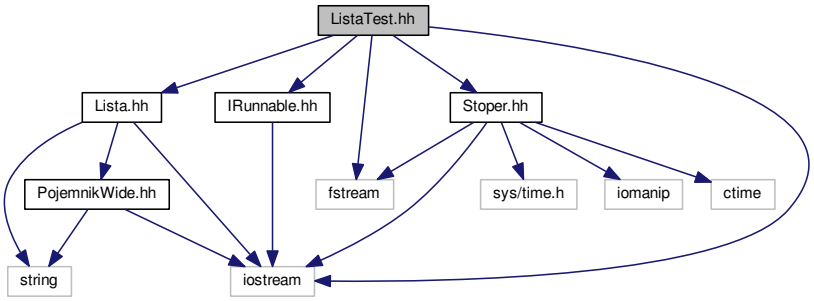


5.8 ListaTest.hh File Reference

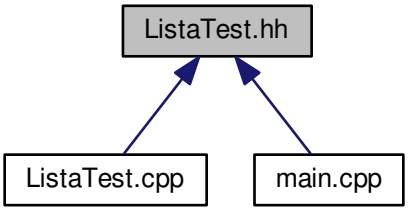
Definicja klasy odpowedzialnej za testowanie "Listy".

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

Include dependency graph for ListaTest.hh:



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



Classes

- class ListaTest

Functions

- bool otworz_plik_odczyt (string nazwapom, fstream &StrmPlikowy)

  otwarcie pliku

5.8.1 Detailed Description

Odpowiedzialna jest za wczytanie danych z pliku, zapisanie ich do listy i znalezienie pozadanego elementu

Definition in file ListaTest.hh.

5.8.2 Function Documentation

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

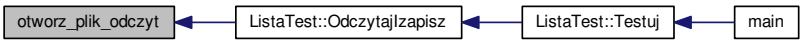
Otwiera plik i tworzy strumien do odczytu

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 ListaTest.cpp.

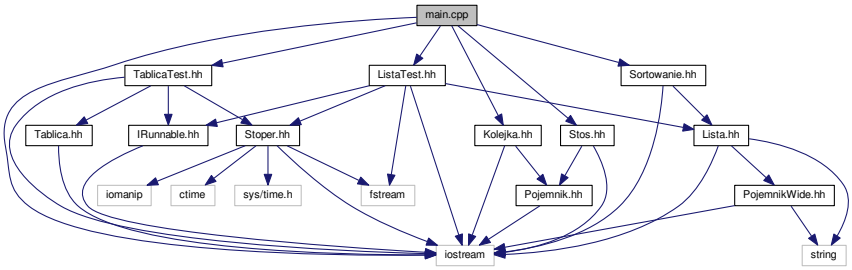
Here is the caller graph for this function:



5.9 main.cpp File Reference

```
#include <iostream>
#include "Kolejka.hh"
#include "Stos.hh"
#include "TablicaTest.hh"
#include "ListaTest.hh"
#include "Sortowanie.hh"
```

Include dependency graph for main.cpp:



Functions

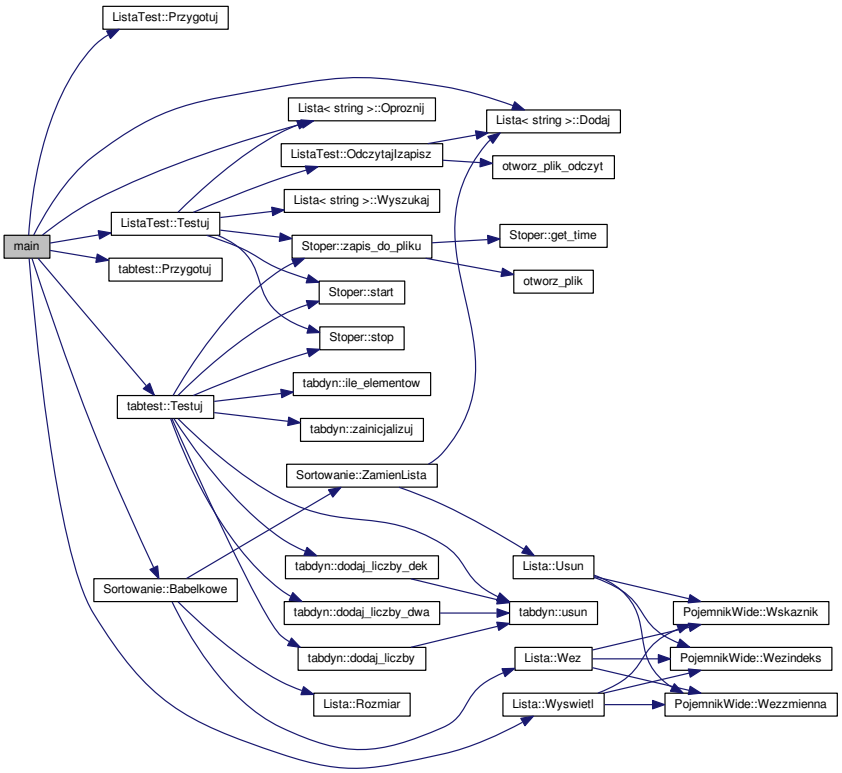
- int main ()

5.9.1 Function Documentation

5.9.1.1 int main ( )

Definition at line 10 of file main.cpp.

Here is the call graph for this function:

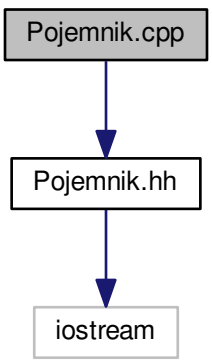


5.10 Pojemnik.cpp File Reference

Definicja metod pojedynczego elementu ADT (Kolejka, Stos)

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

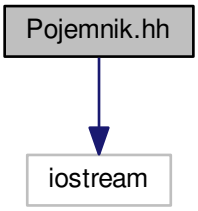
Include dependency graph for Pojemnik.cpp:



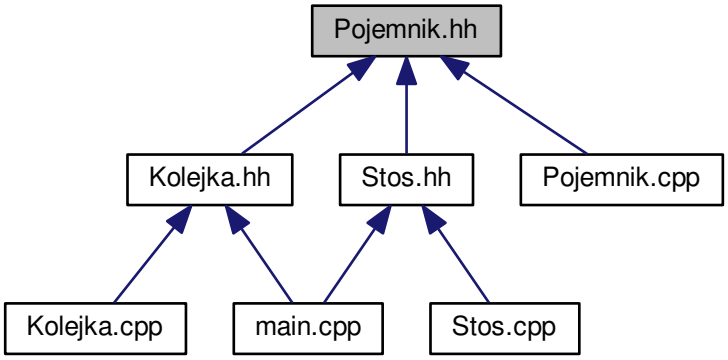
5.11 Pojemnik.hh File Reference

Pelni role pojedynczego elementu ADT (Kolejka, Stos)

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



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



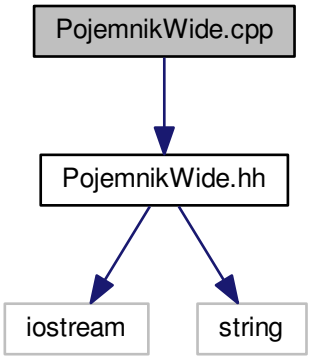
Classes

- class Pojemnik

5.12 PojemnikWide.cpp File Reference

Definicje metod pojedynczego elementu ADT (Lista)

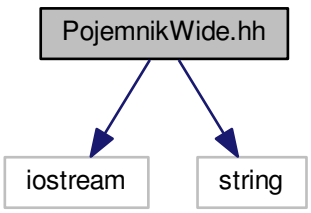
```
#include "PojemnikWide.hh"
```
Include dependency graph for PojemnikWide.cpp:



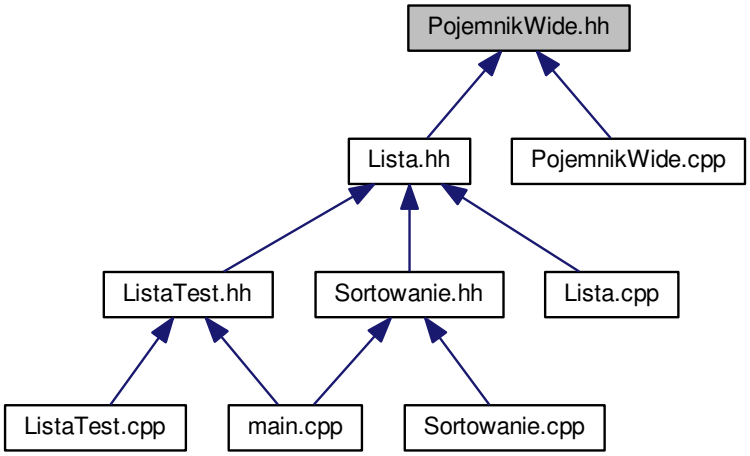
5.13 PojemnikWide.hh File Reference

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

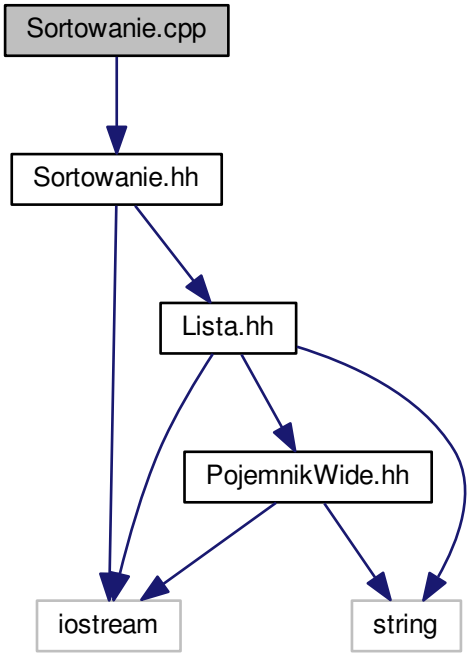
- class PojemnikWide< typ >

5.14 Sortowanie.cpp File Reference

Definicja metod sortowania.

```
#include "Sortowanie.hh"
```

Include dependency graph for Sortowanie.cpp:

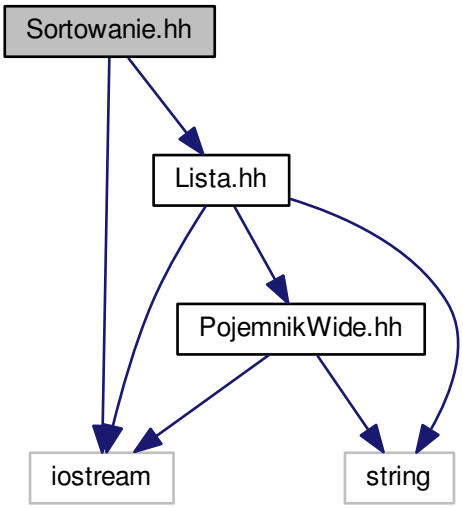


5.15 Sortowanie.hh File Reference

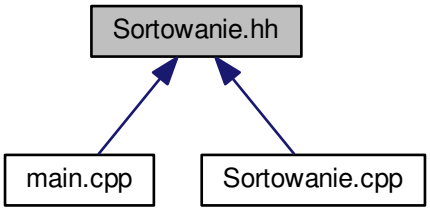
Zawiera metody sortujace.

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

Include dependency graph for Sortowanie.hh:



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



Classes

- class Sortowanie

Macros

- #define ROZMIAR 10

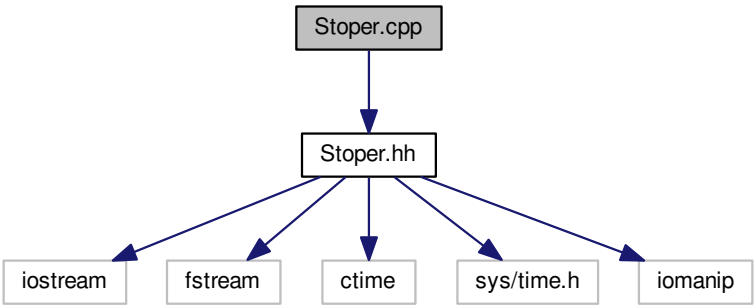
5.15.1 Macro Definition Documentation

5.15.1.1 #define ROZMIAR 10

Definition at line 8 of file Sortowanie.hh.

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*)

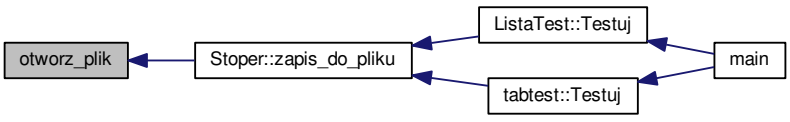
Otwiera plik i tworzy strumien do zapisywania UWAGA: PLIK OTWARTY W TRYBIE DOPISYWANIA

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.

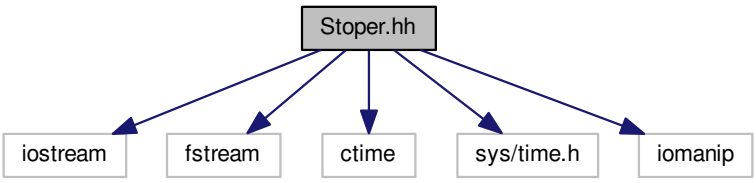
Here is the caller graph for this function:



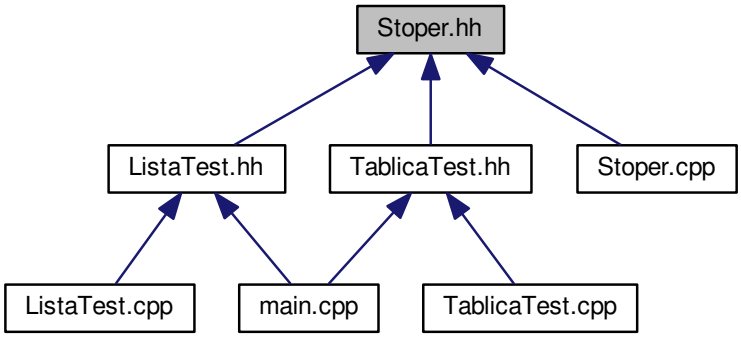
5.17 Stoper.hh File Reference

```
#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*)

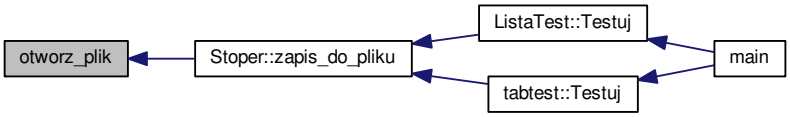
Otwiera plik i tworzy strumien do zapisywania UWAGA: PLIK OTWARTY W TRYBIE DOPISYWANIA

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:

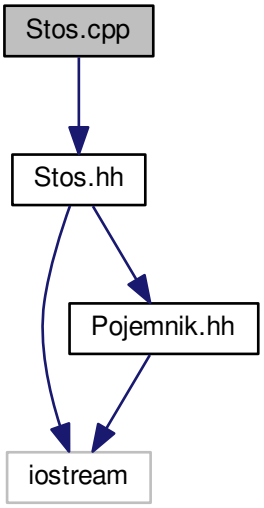


5.18 Stos.cpp File Reference

Definicja metod interface'u ADT- Stos.

```
#include "Stos.hh"
```

Include dependency graph for Stos.cpp:

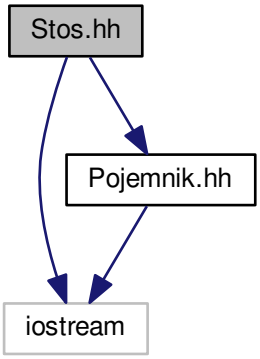


5.19 Stos.hh File Reference

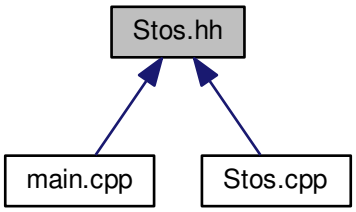
interface abstrakcyjnego typu danych - Stos

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

Include dependency graph for Stos.hh:



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



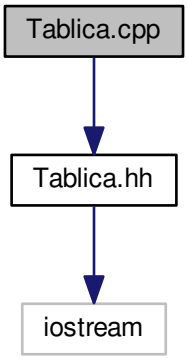
Classes

- class Stos

5.20 Tablica.cpp File Reference

```
#include "Tablica.hh"
```

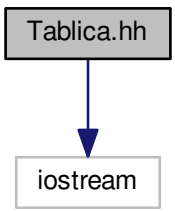
Include dependency graph for Tablica.cpp:



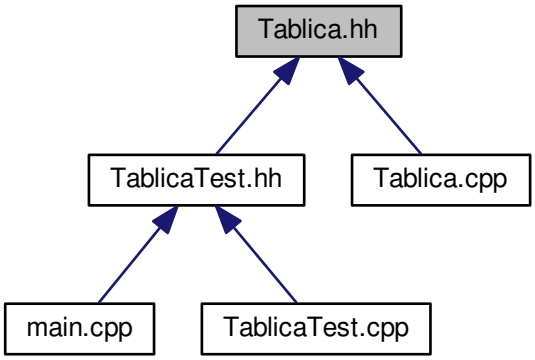
5.21 Tablica.hh File Reference

```
#include <iostream>
```

Include dependency graph for Tablica.hh:



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



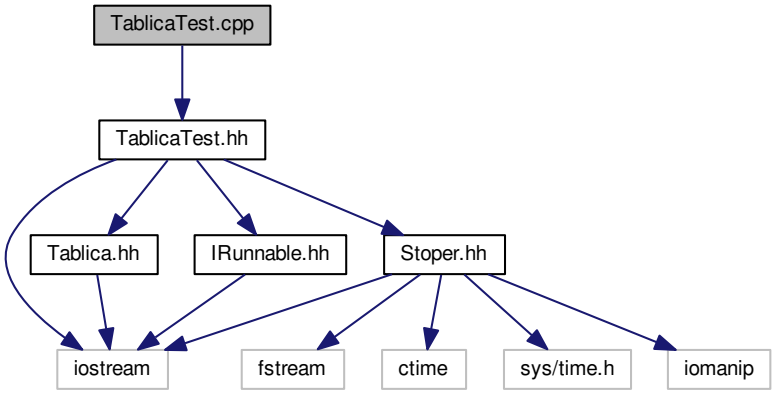
Classes

- class tabdyn

5.22 TablicaTest.cpp File Reference

```
#include "TablicaTest.hh"
```

Include dependency graph for TablicaTest.cpp:

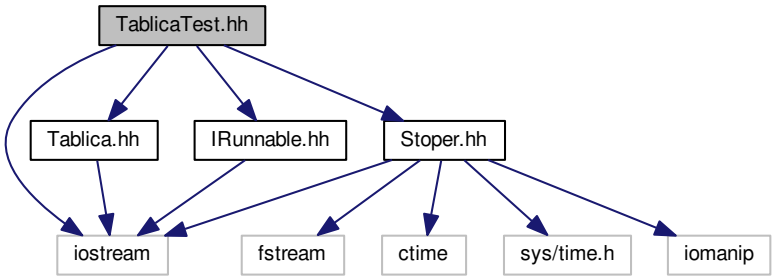


5.23 TablicaTest.hh File Reference

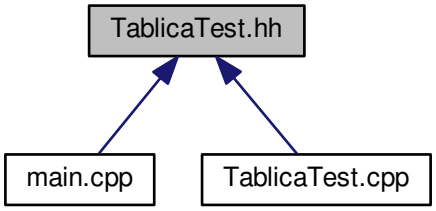
```
#include <iostream>
```

```
#include "Tablica.hh"
#include "Stoper.hh"
#include "IRunnable.hh"
```

Include dependency graph for TablicaTest.hh:



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



Classes

- class tabtest