

PAMSI

6.5

Generated by Doxygen 1.8.9.1

Fri Apr 22 2016 19:56:47

Contents

1	Hierarchical Index	1
1.1	Class Hierarchy	1
2	Class Index	3
2.1	Class List	3
3	File Index	5
3.1	File List	5
4	Class Documentation	7
4.1	lista< typ >::bad_index Class Reference	7
4.1.1	Detailed Description	7
4.2	lista2s< typ >::bad_index Class Reference	7
4.3	tablica< typ >::bad_index Class Reference	7
4.3.1	Detailed Description	7
4.4	tablicatest::bad_index Class Reference	8
4.4.1	Detailed Description	8
4.5	Dictionary Class Reference	8
4.5.1	Member Function Documentation	8
4.5.1.1	countLines	8
4.5.1.2	getWords	8
4.5.1.3	RandomNumber	8
4.5.1.4	RandomWords	9
4.5.1.5	setWords	9
4.6	Drzewo Class Reference	9
4.7	lista2s< typ >::empty Class Reference	10
4.8	tablica< typ >::empty Class Reference	10
4.8.1	Detailed Description	10
4.9	tablicatest::empty Class Reference	10
4.9.1	Detailed Description	10
4.10	lista< typ >::empty Class Reference	10
4.10.1	Detailed Description	11

4.11	iDrzewo< typ > Class Template Reference	11
4.12	inKolejka Class Reference	12
4.13	inLista< typ > Class Template Reference	12
4.14	inStos Class Reference	13
4.15	inTablica< typ > Class Template Reference	14
4.15.1	Detailed Description	14
4.16	iRunnable Class Reference	15
4.17	ITabAsoc Class Reference	15
4.17.1	Detailed Description	16
4.18	ITabZHasz Class Reference	16
4.18.1	Detailed Description	16
4.19	kolejka Class Reference	17
4.19.1	Detailed Description	18
4.20	lista< typ > Class Template Reference	18
4.21	lista2s< typ > Class Template Reference	19
4.22	listatest Class Reference	21
4.23	stoper Class Reference	22
4.24	stos Class Reference	22
4.24.1	Detailed Description	23
4.25	StrListyT< typ > Struct Template Reference	24
4.26	TabAsoc Class Reference	24
4.27	tablica< typ > Class Template Reference	25
4.27.1	Detailed Description	27
4.28	tablicatest Class Reference	27
4.29	TabZHasz Class Reference	29
4.30	TabZHaszTest Class Reference	30
5	File Documentation	31
5.1	inc/inlista.hh File Reference	31
5.1.1	Detailed Description	32
5.2	inc/intablica.hh File Reference	32
5.2.1	Detailed Description	32
5.3	inc/irunnable.hh File Reference	33
5.3.1	Detailed Description	33
5.4	inc/kolejka.hh File Reference	33
5.4.1	Detailed Description	34
5.5	inc/lista.hh File Reference	34
5.5.1	Detailed Description	36
5.6	inc/lista2s.hh File Reference	36
5.6.1	Detailed Description	37

5.7	inc/listatest.hh File Reference	37
5.7.1	Detailed Description	38
5.8	inc/sort.hh File Reference	38
5.8.1	Detailed Description	38
5.9	inc/stos.hh File Reference	38
5.9.1	Detailed Description	39
5.10	inc/strukturalistyT.hh File Reference	39
5.10.1	Detailed Description	40
5.11	inc/tablica.hh File Reference	40
5.11.1	Detailed Description	42
5.12	inc/tablicatest.hh File Reference	42
5.12.1	Detailed Description	42
	Index	43

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

lista< typ >::bad_index	7
lista2s< typ >::bad_index	7
tablica< typ >::bad_index	7
tablicatest::bad_index	8
Dictionary	8
lista2s< typ >::empty	10
tablica< typ >::empty	10
tablicatest::empty	10
lista< typ >::empty	10
iDrzewo< typ >	11
Drzewo	9
inKolejka	12
kolejka	17
inLista< typ >	12
lista< typ >	18
lista2s< typ >	19
inLista< int >	12
lista2s< int >	19
listatest	21
lista< int >	18
inLista< std::string >	12
lista< std::string >	18
inStos	13
stos	22
inTablica< typ >	14
tablica< typ >	25
tablicatest	27
inTablica< int >	14
tablica< int >	25
kolejka	17
stos	22
inTablica< std::string >	14
tablica< std::string >	25
iRunnable	15
listatest	21

tablicatest	27
TabZHaszTest	30
ITabAsoc	15
TabAsoc	24
ITabZHasz	16
TabZHasz	29
stoper	22
StrListyT< typ >	24
StrListyT< int >	24

Chapter 2

Class Index

2.1 Class List

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

lista< typ >::bad_index	Obiekt zwracany podczas wystapienia wyjatku wystepujacego gdy nastapi odwołanie do danych o złym indeksie	7
lista2s< typ >::bad_index	7
tablica< typ >::bad_index	Obiekt zwracany podczas wystapienia wyjatku wystepujacego gdy nastapi odwołanie do danych o złym indeksie	7
tablicatest::bad_index	Obiekt zwracany podczas wystapienia wyjatku wystepujacego gdy nastapi proba odzialywania na dane ktorych kontener nie posiada	8
Dictionary	8
Drzewo	9
lista2s< typ >::empty	10
tablica< typ >::empty	Obiekt zwracany podczas wystapienia wyjatku wystepujacego gdy nastapi proba odzialywania na dane ktorych kontener nie posiada	10
tablicatest::empty	Metoda sprawdza czy kontener jest pusty	10
lista< typ >::empty	Obiekt zwracany podczas wystapienia wyjatku wystepujacego gdy nastapi proba odzialywania na dane ktorych kontener nie posiada	10
iDrzewo< typ >	11
inKolejka	12
inLista< typ >	12
inStos	13
inTablica< typ >	Class inTablica- interfejs do tablicy dynamicznie realokujacej miejsce	14
iRunnable	15
ITabAsoc	/file /brief plik zawiera interfejs wykorzystywany do tablicy asocjacyjnej	15
ITabZHasz	/file /brief plik zawiera interfejs wykorzystywany do tablicy z haszowaniem	16
kolejka	Klasa kolejka - jeden z abstrakcyjnych typow danych Klasa ma w swoim skladzie metody sluzace do zarzadzania kolejka, wszystkie operacje poza dodaniem nowej danej do pustej kolejki zgloszaja wyjatek	17
lista< typ >	18
lista2s< typ >	19

listatest	21
stoper	22
stos	
Klasa stos - jeden z abstrakcyjnych typow danych	22
StrListyT< typ >	24
TabAsoc	24
tablica< typ >	
Klasa tablica Klasa ma w swoim skladzie metody sluzace do zarzadzania tablica dynamiczna uzywane przez inne typy danych, moze byc tez uzywana samodzielnie	25
tablicatest	27
TabZHasz	29
TabZHaszTest	30

Chapter 3

File Index

3.1 File List

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

inc/ Dictionary.h	??
inc/ drzewo.hh	??
inc/ idrzewo.hh	??
inc/ inkolejka.hh	??
inc/ inlista.hh	
Plik zawiera interfejs uzywany w listach	31
inc/ instos.hh	??
inc/ intablica.hh	
Plik zawiera interfejs do klasy tablica	32
inc/ irunnable.hh	
Zawira interfejs do testowania zaimplementowanych algorytmow	33
inc/ itabasoc.hh	??
inc/ itabzhasz.hh	??
inc/ kolejka.hh	
Plik zawiera definicje klasy kolejka	33
inc/ lista.hh	
Plik zawiera klase lista, ktora jest oparta na jednowymiarowej tablicy dynamicznej	34
inc/ lista2s.hh	
Plik zawiera szablon klasy lista2s , ktora jest zbudowana jak klasyczna lista	36
inc/ listatest.hh	
Plik zawiera klase listatest odpowiadajaca za testowanie listy	37
inc/ sort.hh	
Plik zawiera funkcje z metodami sortowania	38
inc/ stoper.hh	??
inc/ stos.hh	
Plik zawiera definicje klasy stos	38
inc/ strukturalistyT.hh	
Plik zawiera strukture potrzebna do dzialania listy	39
inc/ tabasoc.hh	??
inc/ tablica.hh	
Plik zawiera klase tablica	40
inc/ tablicatest.hh	
Plik zawiera klase tablicatest	42
inc/ tabzhasz.hh	??
inc/ tabzhasztest.hh	??

Chapter 4

Class Documentation

4.1 `lista< typ >::bad_index` Class Reference

obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi odwołanie do danych o złym indeksie

```
#include <lista.hh>
```

4.1.1 Detailed Description

```
template<class typ>class lista< typ >::bad_index
```

obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi odwołanie do danych o złym indeksie

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

- [inc/lista.hh](#)

4.2 `lista2s< typ >::bad_index` Class Reference

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

- [inc/lista2s.hh](#)

4.3 `tablica< typ >::bad_index` Class Reference

obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi odwołanie do danych o złym indeksie

```
#include <tablica.hh>
```

4.3.1 Detailed Description

```
template<class typ>class tablica< typ >::bad_index
```

obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi odwołanie do danych o złym indeksie

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

- [inc/tablica.hh](#)

4.4 tablicatest::bad_index Class Reference

obiekt zwracany podczas wystapienia wyjatku wystepujacego gdy nastapi proba odzialywania na dane ktorych kontener nie posiada

```
#include <tablicatest.hh>
```

4.4.1 Detailed Description

obiekt zwracany podczas wystapienia wyjatku wystepujacego gdy nastapi proba odzialywania na dane ktorych kontener nie posiada

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

- inc/[tablicatest.hh](#)

4.5 Dictionary Class Reference

Public Member Functions

- void [setWords](#) ()
- string [getWords](#) (int i)
- int [countLines](#) ()
- string [RandomWords](#) ()
- int [RandomNumber](#) ()

4.5.1 Member Function Documentation

4.5.1.1 int Dictionary::countLines ()

funkcja zliczajaca linie w pliku (ilosc slow w slowniku)

Returns

lines

4.5.1.2 string Dictionary::getWords (int i)

funkcja zwracajaca slowo z tablicy z wyrazami ze slownika

Parameters

<i>i</i>	- numer indeksu zwracanego slowa
----------	----------------------------------

Returns

words_[i]

4.5.1.3 int Dictionary::RandomNumber ()

funkcja generujaca losowe liczby

Returns

4.5.1.4 string Dictionary::RandomWords ()

funkcja generująca losowe słowa na podstawie słownika

Returns

4.5.1.5 void Dictionary::setWords ()

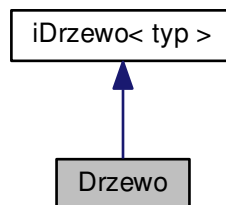
funkcja wczytująca słownik do tablicy

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

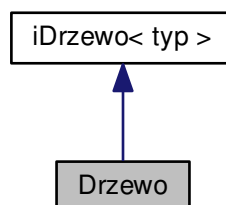
- inc/Dictionary.h
- src/Dictionary.cpp

4.6 Drzewo Class Reference

Inheritance diagram for Drzewo:



Collaboration diagram for Drzewo:



Public Member Functions

- void **add** (int)

- void **get** (int)

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

- [inc/drzewo.hh](#)

4.7 lista2s< typ >::empty Class Reference

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

- [inc/lista2s.hh](#)

4.8 tablica< typ >::empty Class Reference

obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi próba oddziaływania na dane których kontener nie posiada

```
#include <tablica.hh>
```

4.8.1 Detailed Description

```
template<class typ>class tablica< typ >::empty
```

obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi próba oddziaływania na dane których kontener nie posiada

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

- [inc/tablica.hh](#)

4.9 tablicatest::empty Class Reference

metoda sprawdza czy kontener jest pusty

```
#include <tablicatest.hh>
```

4.9.1 Detailed Description

metoda sprawdza czy kontener jest pusty

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

- [inc/tablicatest.hh](#)

4.10 lista< typ >::empty Class Reference

obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi próba oddziaływania na dane których kontener nie posiada

```
#include <lista.hh>
```


4.10.1 Detailed Description

```
template<class typ>class lista< typ >::empty
```

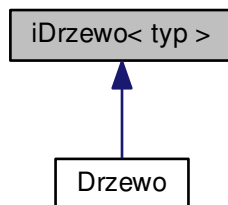
obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi próba oddziaływania na dane których kontener nie posiada

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

- [inc/lista.hh](#)

4.11 iDrzewo< typ > Class Template Reference

Inheritance diagram for iDrzewo< typ >:



Public Member Functions

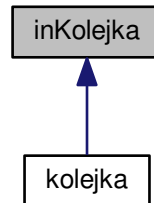
- virtual void **add** (typ)=0
- virtual typ **get** (int)=0

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

- [inc/idrzewo.hh](#)

4.12 inKolejka Class Reference

Inheritance diagram for inKolejka:



Public Member Functions

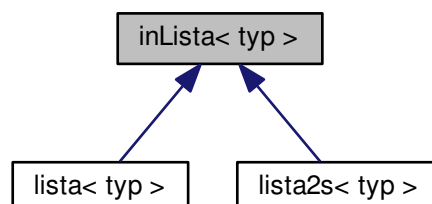
- virtual void [remove](#) ()=0
usuwa pierwsza dana w kolejce
- virtual void [add](#) (int)=0
dodaje dana na koniec kolejki
- virtual int [last](#) ()=0
zwraca wartosc pierwszej danej
- virtual int [first](#) ()=0
zwraca wartosc ostatniej danej

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

- inc/inkolejka.hh

4.13 inLista< typ > Class Template Reference

Inheritance diagram for inLista< typ >:



Public Member Functions

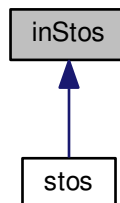
- virtual void `remove` (int)=0
usuwa dana o podanym indeksie, a nastepnie przestawia zmienne
- virtual void `remove` ()=0
usuwa ostatnia dana w kolejce
- virtual void `add` (typ, int)=0
dodaje dana na miejscu o podanym indeksie, jezeli nie moze byc tam wstawiona zglosi wyjatek, gdy miejsce jest zajete przestawia dane
- virtual void `add` (typ)=0
dodaje dana na koniec listy
- virtual typ `get` (int)=0
zwraca wartosc na podanym indeksie, jezeli nie moze tego zrobic, zwraci wyjatek
- virtual bool `isEmpty` ()=0
zwraca wartosc true gdy lista jest pusta lub false gdy jest na niej przynajmniej jeden element
- virtual int `size` ()=0
zwraca liczbe elementow na stosie

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

- inc/inlista.hh

4.14 inStos Class Reference

Inheritance diagram for inStos:



Public Member Functions

- virtual void `remove` ()=0
metoda `remove()`- nie przyjmuje wartosci, usuwa najwyzej polozona na stosie dana
- virtual void `add` (int)=0
metoda `add(int)`- przyjmuje wartosc int, bedaca nowa liczba do dodania na szczyt stosu
- virtual int `get` ()=0
metoda `get` bez parametru- zwraca wartosc ostatniej liczby na stosie

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

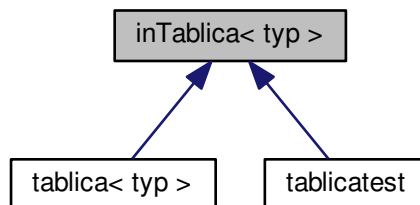
- inc/instos.hh

4.15 inTablica< typ > Class Template Reference

class inTablica- interfejs do tablicy dynamicznie realokujacej miejsce

```
#include <intablica.hh>
```

Inheritance diagram for inTablica< typ >:



Public Member Functions

- virtual void [push](#) (typ)=0
metoda dodajaca dana na ostatnie wolne miejsce tablicy
- virtual void [przypisz](#) (typ, int)=0
przypisuje podana dana na konkretnym miejscu tablicy, pierwszym parametrem jest dana, a drugim miejsce na ktorym ma byc zapisana
- virtual void [zdejmij](#) (int)=0
usuwa podany element tablicy
- virtual int [size](#) ()=0
metoda zwraca ilosc zaalokowanego miejsca
- virtual int [ind](#) ()=0
metoda zwraca indeks na ktorym jest ostatnia dana w kontenerze
- virtual bool [isEmpty](#) ()=0
metoda sprawdza czy kontener jest pusty

4.15.1 Detailed Description

```
template<class typ>class inTablica< typ >
```

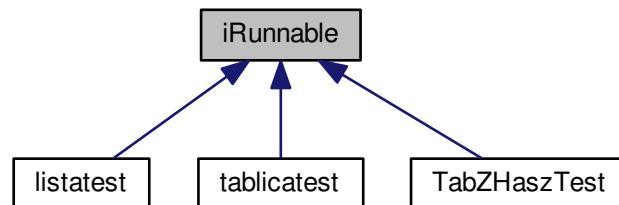
class inTablica- interfejs do tablicy dynamicznie realokujacej miejsce

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

- inc/[intablica.hh](#)

4.16 iRunnable Class Reference

Inheritance diagram for iRunnable:



Public Member Functions

- virtual void **run** (int)=0
uruchamia testy dla zadanego algorytmu
- void **prepare** (int, int)

Protected Attributes

- int **ileOkrazen** =10
- int **ileDanych** =100000000

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

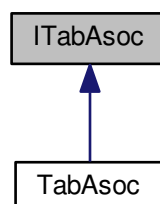
- inc/[irunnable.hh](#)

4.17 ITabAsoc Class Reference

/file /brief plik zawiera interfejs wykorzystywany do tablicy asocjacyjnej

```
#include <itabasoc.hh>
```

Inheritance diagram for ITabAsoc:



Public Member Functions

- virtual int [search](#) (std::string)=0
szukanie wartosci dla frazy
- virtual void [insert](#) (std::string, int)=0
wstawienie wyrazu, o przypisanej pewnej wartosci calkowitej

4.17.1 Detailed Description

/file /brief plik zawiera interfejs wykorzystywany do tablicy asocjacyjnej

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

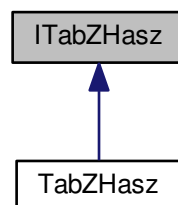
- inc/itabasoc.hh

4.18 ITabZHasz Class Reference

/file /brief plik zawiera interfejs wykorzystywany do tablicy z haszowaniem

```
#include <itabzhasz.hh>
```

Inheritance diagram for ITabZHasz:



Public Member Functions

- virtual [TabAsoc](#) & [operator\[\]](#) (const std::string)=0
przeciazenie operatora przypisuje ciagowi znakow liczbe

4.18.1 Detailed Description

/file /brief plik zawiera interfejs wykorzystywany do tablicy z haszowaniem

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

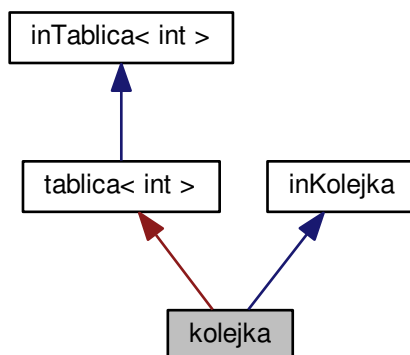
- inc/itabzhasz.hh

4.19 kolejka Class Reference

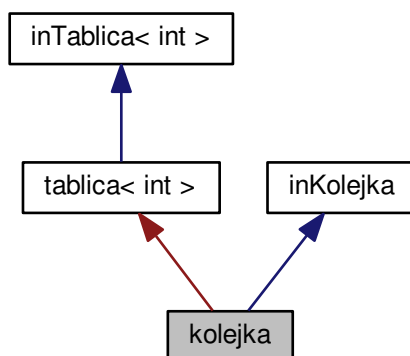
Klasa kolejka - jeden z abstrakcyjnych typow danych Klasa ma w swoim skladzie metody sluzace do zarzadzania kolejka, wszystkie operacje poza dodaniem nowej danej do pustej kolejki zgłaszają wyjątek.

```
#include <kolejka.hh>
```

Inheritance diagram for kolejka:



Collaboration diagram for kolejka:



Public Member Functions

- void `remove` ()
usuwa pierwsza dana w kolejce
- void `add` (int)
dodaje dana na koniec kolejki
- int `first` ()

- zwraca wartosc ostatniej danej*
 - int [last](#) ()*zwraca wartosc pierwszej danej*

4.19.1 Detailed Description

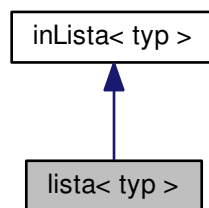
Klasa kolejka - jeden z abstrakcyjnych typow danych Klasa ma w swoim skladzie metody sluzace do zarzadzania kolejka, wszystkie operacje poza dodaniem nowej danej do pustej kolejki zgłaszają wyjątek.

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

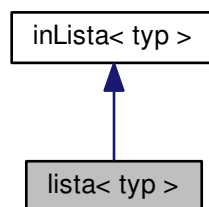
- inc/[kolejka.hh](#)
- src/[kolejka.cpp](#)

4.20 [lista< typ >](#) Class Template Reference

Inheritance diagram for [lista< typ >](#):



Collaboration diagram for [lista< typ >](#):



Classes

- class [bad_index](#)
 - obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi odwołanie do danych o złym indeksie*

- class [empty](#)

obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi próba oddziaływania na dane których kontener nie posiada

Public Member Functions

- [lista](#) ()
konstruktor bezparametryczny
- void [remove](#) (int)
usuwa dana o podanym indeksie, a następnie przestawia zmienne
- void [remove](#) ()
usuwa ostatnia dana w kolejce
- void [add](#) (typ, int)
dodaje dana na miejscu o podanym indeksie, jezeli nie moze byc tam wstawiona zglosi wyjatek, gdy miejsce jest zajete przestawia dane
- void [add](#) (typ)
dodaje dana na koniec listy
- typ [get](#) (int)
zwraca wartosc na podanym indeksie, jezeli nie moze tego zrobic, zwroci wyjatek
- bool [isEmpty](#) ()
zwraca wartosc true gdy lista jest pusta lub false gdy jest na niej przynajmniej jeden element
- int [size](#) ()
zwraca liczbe elementow na stosie

Protected Attributes

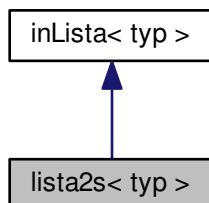
- [tablica](#)< typ > * **Tab**

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

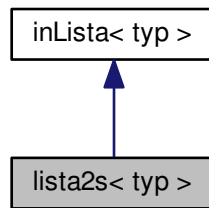
- inc/[lista.hh](#)

4.21 lista2s< typ > Class Template Reference

Inheritance diagram for lista2s< typ >:



Collaboration diagram for lista2s< typ >:



Classes

- class [bad_index](#)
- class [empty](#)

Public Member Functions

- [lista2s](#) ()
konstruktor bezparametryczny
- void [remove](#) (int)
usuwa dana o podanym indeksie, a nastepnie przestawia zmienne
- void [remove](#) ()
usuwa ostatnia dana w kolejce
- void [add](#) (typ, int)
dodaje dana na miejscu o podanym indeksie, jezeli nie moze byc tam wstawiona zglosi wyjatek, gdy miejsce jest zajete przestawia dane
- void [add](#) (typ)
dodaje dana na koniec listy
- typ [get](#) (int)
zwraca wartosc na podanym indeksie, jezeli nie moze tego zrobic, zwraci wyjatek
- bool [isEmpty](#) ()
zwraca wartosc true gdy lista jest pusta lub false gdy jest na niej przynajmniej jeden element
- int [size](#) ()
zwraca liczbe elementow na stosie
- int [find](#) (typ)

Protected Attributes

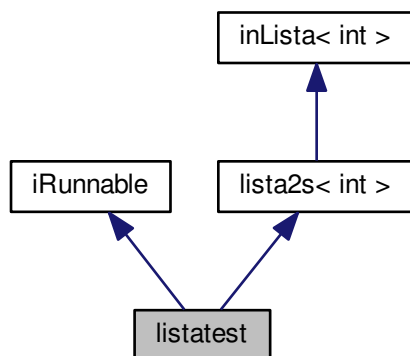
- [StrListyT](#)< typ > * **FIRST**
- int **rozmiar** =0

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

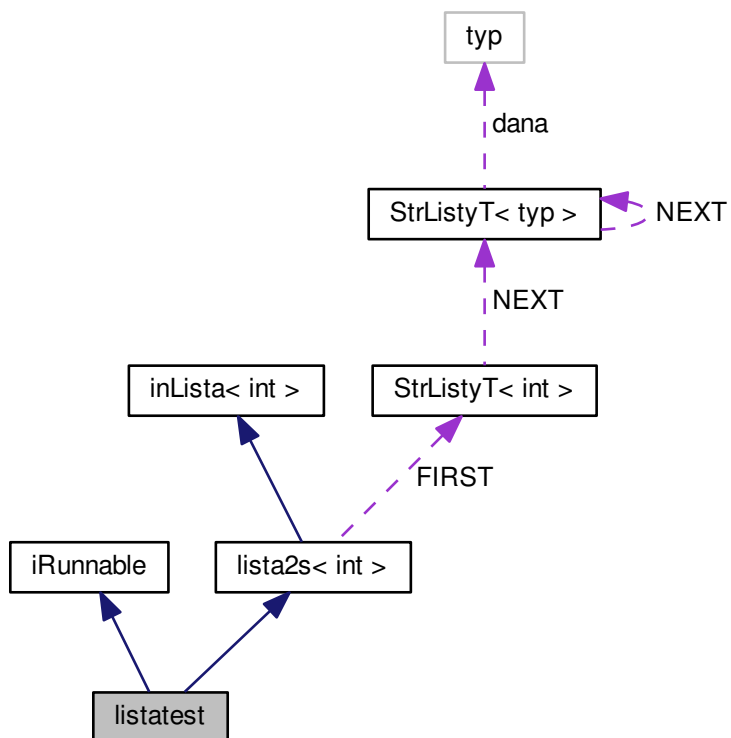
- inc/[lista2s.hh](#)

4.22 listatest Class Reference

Inheritance diagram for listatest:



Collaboration diagram for listatest:



Public Member Functions

- void **run** (int)
uruchamia testy dla zadanego algorytmu

Additional Inherited Members

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

- inc/listatest.hh
- src/listatest.cpp

4.23 stoper Class Reference

Public Member Functions

- void **start** ()
- void **stop** ()
- std::chrono::duration< double > **getElapsedTime** ()
- std::chrono::duration< double > **getTime** ()
- bool **dumpToFile** (std::string)

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

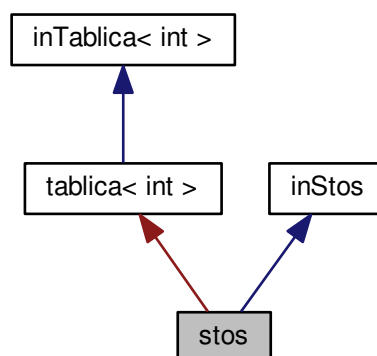
- inc/stoper.hh
- src/stoper.cpp

4.24 stos Class Reference

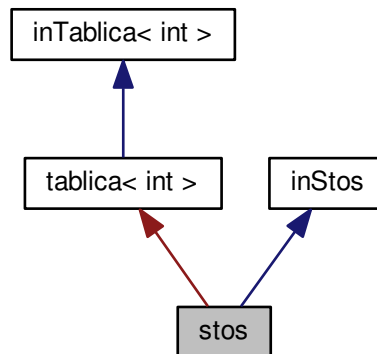
Klasa stos - jeden z abstrakcyjnych typow danych.

```
#include <stos.hh>
```

Inheritance diagram for stos:



Collaboration diagram for stos:



Public Member Functions

- `stos ()`

konstruktor bezparametryczny

- `void remove ()`

metoda `remove()`- nie przyjmuje wartosci, usuwa najwyzej polozona na stosie dana

- `void add (int)`

metoda `add(int)`- przyjmuje wartosc int, bedaca nowa liczba do dodania na szczyt stosu

- `int get ()`

metoda `get` bez parametru- zwraca wartosc ostatniej liczby na stosie

4.24.1 Detailed Description

Klasa `stos` - jeden z abstrakcyjnych typow danych.

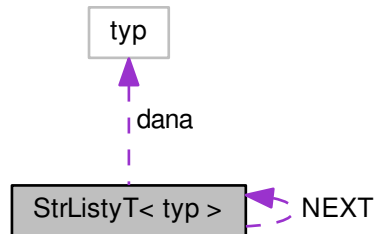
Klasa ma w swoim skladzie metody sluzace do zarzadzania stosem, mozliwe jest jedynie manipulowanie najwyzej polozonym elementem na stosie, inne operacje poza dodaniem nowej danej do niego na pustym stosie zgłaszają wyjątek

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

- `inc/stos.hh`
- `src/stos.cpp`

4.25 StrListyT< typ > Struct Template Reference

Collaboration diagram for StrListyT< typ >:



Public Attributes

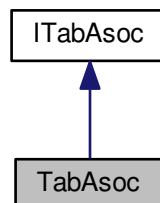
- `typ dana`
- `StrListyT * NEXT`

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

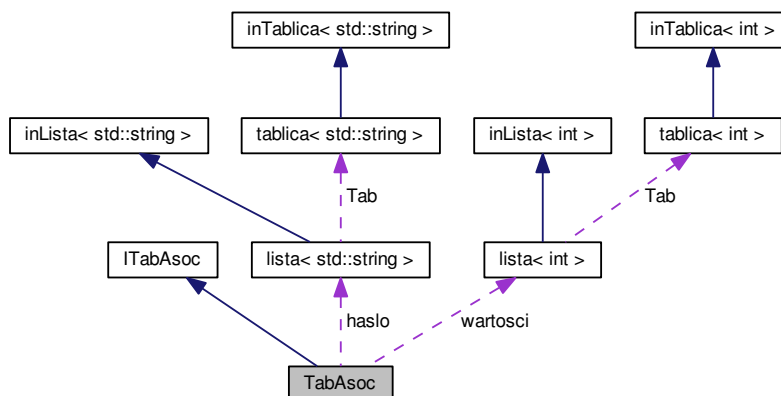
- `inc/strukturalistyT.hh`

4.26 TabAsoc Class Reference

Inheritance diagram for TabAsoc:



Collaboration diagram for TabAsoc:



Public Member Functions

- `int search (std::string)`
szukanie wartosci dla frazy
- `void insert (std::string, int)`
wstawienie wyrazu, o przypisanej pewnej wartosci calkowitej

Protected Attributes

- `lista< std::string > * haslo`
- `lista< int > * wartosci`

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

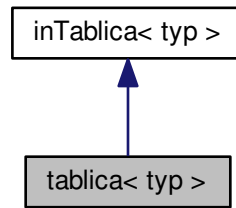
- `inc/tabasoc.hh`
- `src/tabasoc.cpp`

4.27 `tablica< typ >` Class Template Reference

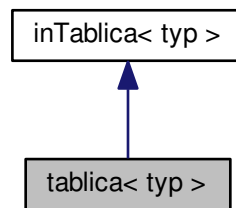
Klasa `tablica` Klasa ma w swoim skladzie metody sluzace do zarzadzania tablica dynamiczna uzywane przez inne typy danych, moze byc tez uzywana samodzielnie.

```
#include <tablica.hh>
```

Inheritance diagram for `tablica< typ >`:



Collaboration diagram for `tablica< typ >`:



Classes

- class `bad_index`
obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi odwołanie do danych o złym indeksie
- class `empty`
obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi próba oddziaływania na dane których kontener nie posiada

Public Member Functions

- void `zmniejsz` ()
zmniejsza ilość zaalokowanej pamięci
- `tablica` ()
konstruktor bezparametryczny, tworzy tablice o 10 polach
- `tablica` (int n)
konstruktor, tworzy tablice o podanej przez użytkownika ilości pol
- typ `get` (int n)
- void `push` (typ)
metoda dodająca dana na ostatnie wolne miejsce tablicy
- void `przypisz` (typ, int)

przypisuje podana dana na konkretnym miejscu tablicy, pierwszym parametrem jest dana, a drugim miejsce na którym ma byc zapisana

- void [zdejmij](#) (int)

usuwa podany element tablicy

- void [bubblesort](#) ()

sortowanie bombelkowe

- int [size](#) ()

metoda zwraca ilosc zaalokowanego miejsca

- int [ind](#) ()

metoda zwraca ideks na ktorem jest ostatnia dana w kontenerze

- bool [isEmpty](#) ()

metoda sprawdza czy kontener jest pusty

Public Attributes

- int [ile_elem](#)

4.27.1 Detailed Description

`template<class typ>class tablica< typ >`

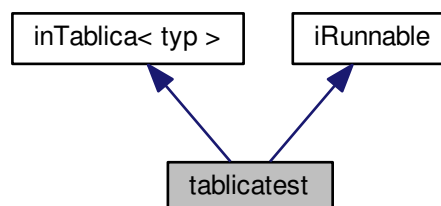
Klasa tablica Klasa ma w swoim skladzie metody sluzace do zarzadzania tablica dynamiczna uzywane przez inne typy danych, moze byc tez uzywana samodzielnie.

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

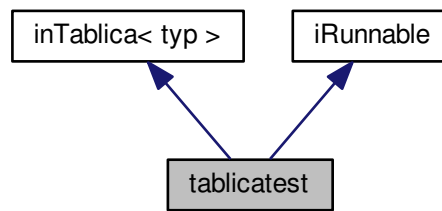
- inc/[tablica.hh](#)
- src/tablicatest.cpp

4.28 tablicatest Class Reference

Inheritance diagram for tablicatest:



Collaboration diagram for tablicatest:



Classes

- class `bad_index`
obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi próba oddziaływania na dane których kontener nie posiada
- class `empty`
metoda sprawdza czy kontener jest pusty

Public Member Functions

- void `zmniejsz` ()
- `tablica` ()
zwiększa ilość zaalokowanej pamięci
- `tablica` (int n)
konstruktor bezparametryczny, tworzy tablice o 10 polach
- void `push` (int)
konstruktor, tworzy tablice o podanej przez użytkownika ilości pol
- void `przypisz` (int, int)
metoda dodająca dana na ostatnie wolne miejsce tablicy
- void `zdejmij` (int)
przypisuje podana dana na konkretnym miejscu tablicy, pierwszym parametrem jest dana, a drugim miejsce na którym ma być zapisana
- void `bubblesort` ()
usuwa ostatni element tablicy
- int `size` ()
sortowanie bombelkowe
- int `ind` ()
metoda zwraca ilość zaalokowanego miejsca
- bool `isEmpty` ()
metoda zwraca indeks na którym jest ostatnia dana w kontenerze
- void `run` ()
obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi odwołanie do danych o złym indeksie

Public Attributes

- int `ile_elem`

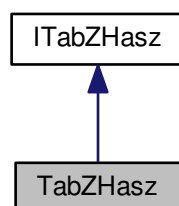
Additional Inherited Members

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

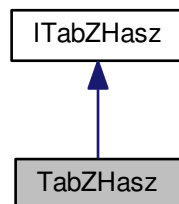
- [inc/tablicatest.hh](#)
- [src/tablicatest.cpp](#)

4.29 TabZHasz Class Reference

Inheritance diagram for TabZHasz:



Collaboration diagram for TabZHasz:



Public Member Functions

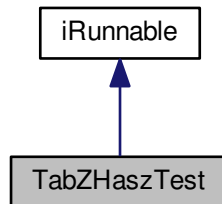
- [TabZHasz](#) ()
konstruktor bezparametryczny, zakłada ze liczba kontenerow = 10
- [TabZHasz](#) (int ile)
konstruktor z jednym parametrem określającym ilość kontenerow z danymi
- [TabAsoc](#) & [operator\[\]](#) (std::string)
przeciążenie operatora przypisuje ciągowi znaków liczbę

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

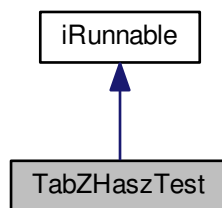
- [inc/tabzhasz.hh](#)
- [src/tabzhasz.cpp](#)

4.30 TabZHaszTest Class Reference

Inheritance diagram for TabZHaszTest:



Collaboration diagram for TabZHaszTest:



Public Member Functions

- void [run](#) (int)
uruchamia testy dla zadanego algorytmu

Additional Inherited Members

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

- inc/tabzhasztest.hh
- src/tabzhasztest.cpp

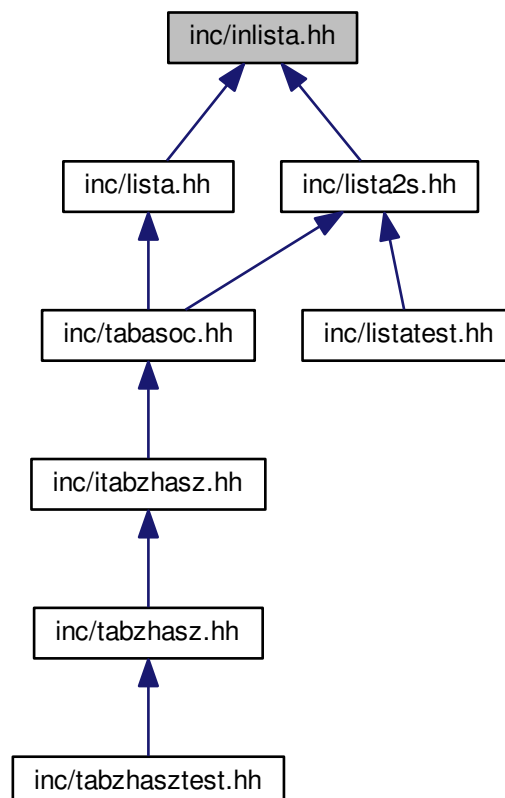
Chapter 5

File Documentation

5.1 inc/inlista.hh File Reference

plik zawiera interfejs uzywany w listach

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



Classes

- class [inLista< typ >](#)

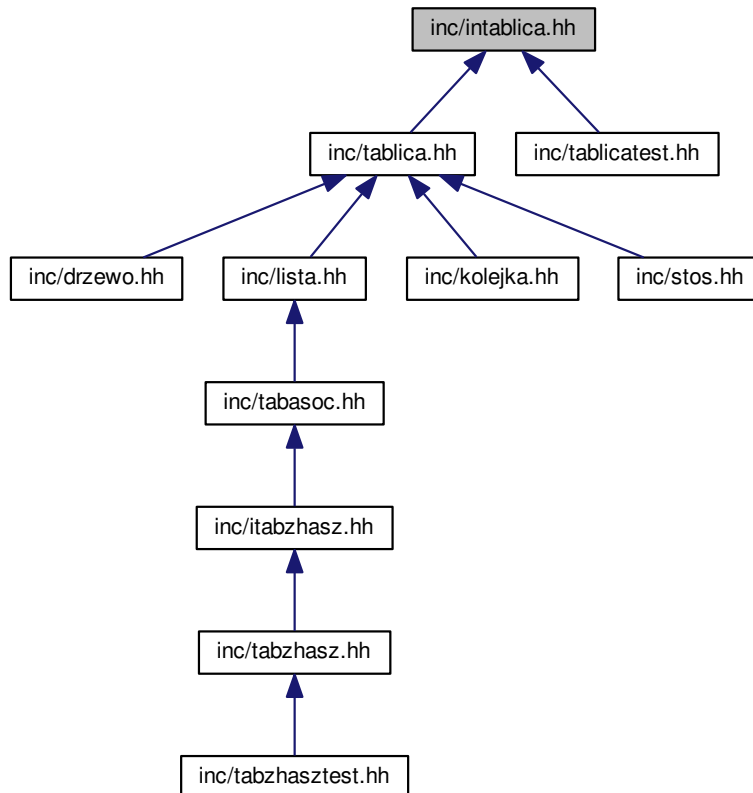
5.1.1 Detailed Description

plik zawiera interfejs uzywany w listach

5.2 inc/intablica.hh File Reference

plik zawiera interfejs do klasy tablica

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



Classes

- class [inTablica< typ >](#)
class inTablica- interfejs do tablicy dynamicznie realokujacej miejsce

5.2.1 Detailed Description

plik zawiera interfejs do klasy tablica

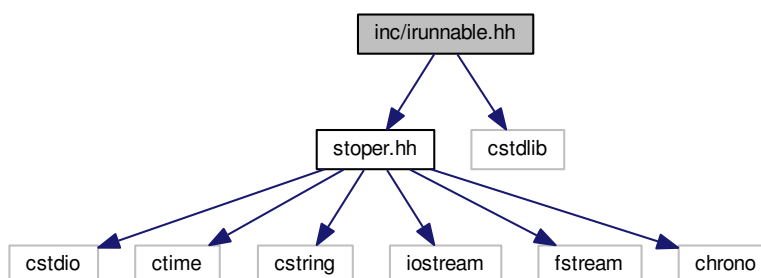
5.3 inc/irunnable.hh File Reference

zawiera interfejs do testowania zaimplementowanych algorytmow

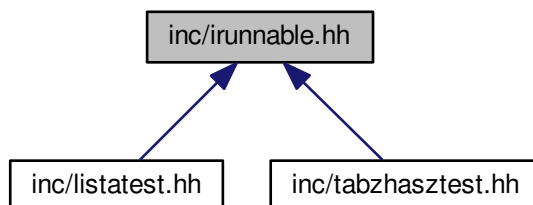
```
#include "stoper.hh"
```

```
#include <cstdlib>
```

Include dependency graph for irunnable.hh:



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



Classes

- class [iRunnable](#)

5.3.1 Detailed Description

zawiera interfejs do testowania zaimplementowanych algorytmow

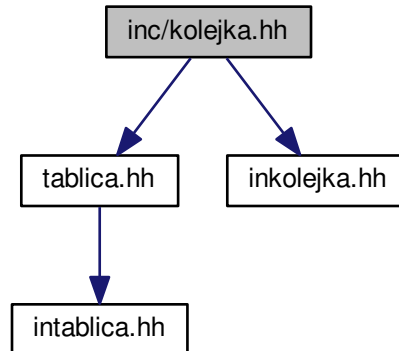
5.4 inc/kolejka.hh File Reference

plik zawiera definicje klasy kolejka

```
#include "tablica.hh"
```

```
#include "inkolejka.hh"
```

Include dependency graph for kolejka.hh:



Classes

- class `kolejka`

Klasa kolejka - jeden z abstrakcyjnych typow danych Klasa ma w swoim skladzie metody sluzace do zarzadzania kolejka, wszystkie operacje poza dodaniem nowej danej do pustej kolejki zgloszaja wyjatek.

5.4.1 Detailed Description

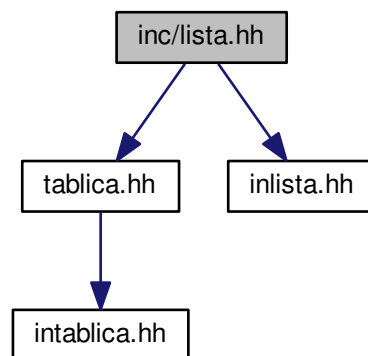
plik zawiera definicje klasy kolejka

5.5 inc/lista.hh File Reference

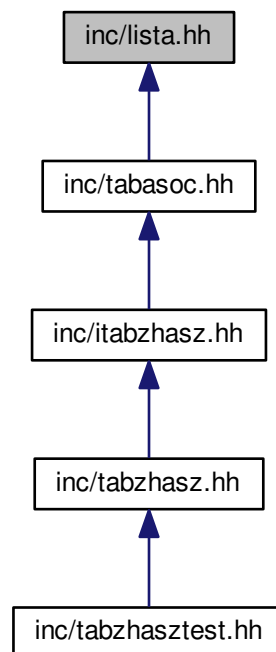
plik zawiera klase lista, ktora jest oparta na jednowymiarowej tablicy dynamicznej

```
#include "tablica.hh"
#include "inlista.hh"
```


Include dependency graph for lista.hh:



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



Classes

- class `lista< typ >`
- class `lista< typ >::empty`

obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi próba oddziaływania na dane których kontener nie posiada

- class `lista< typ >::bad_index`

obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi odwołanie do danych o złym indeksie

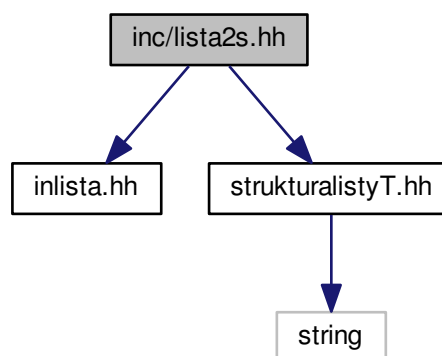
5.5.1 Detailed Description

plik zawiera klasę `lista`, która jest oparta na jednowymiarowej tablicy dynamicznej

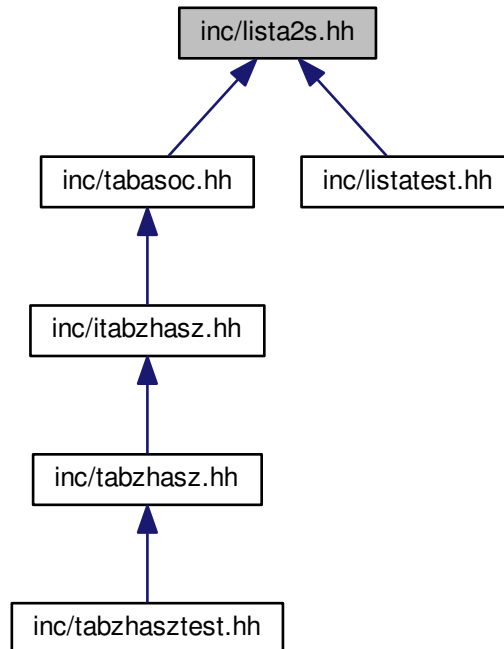
5.6 inc/lista2s.hh File Reference

plik zawiera szablon klasy `lista2s`, która jest zbudowana jak klasyczna lista

```
#include "inlista.hh"
#include "strukturalistyT.hh"
Include dependency graph for lista2s.hh:
```



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



Classes

- class [lista2s< typ >](#)
- class [lista2s< typ >::bad_index](#)
- class [lista2s< typ >::empty](#)

5.6.1 Detailed Description

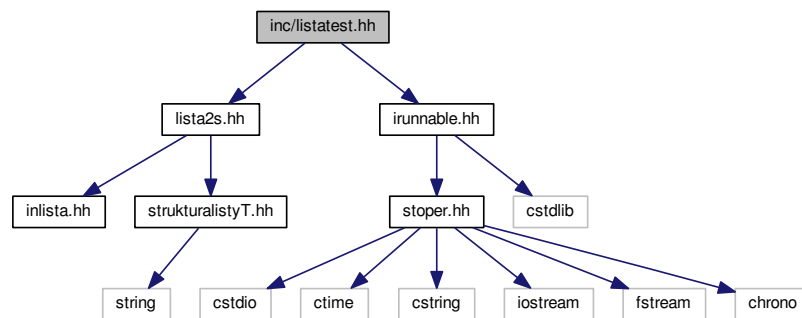
plik zawiera szablon klasy [lista2s](#), która jest zbudowana jak klasyczna lista

5.7 inc/listatest.hh File Reference

plik zawiera klasę listatest odpowiadającą za testowanie listy

```
#include "lista2s.hh"
#include "irunnable.hh"
```

Include dependency graph for listatest.hh:



Classes

- class [listatest](#)

5.7.1 Detailed Description

plik zawiera klase listatest odpowiadajaca za testowanie listy

5.8 inc/sort.hh File Reference

plik zawiera funkcje z metodami sortowania

Functions

- void **bubblesort** (int tab[], int size)

5.8.1 Detailed Description

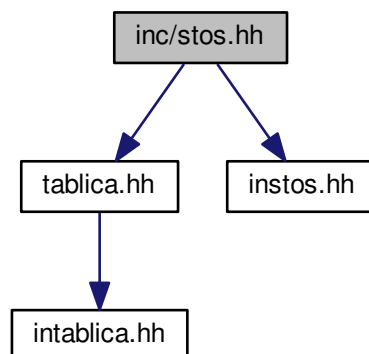
plik zawiera funkcje z metodami sortowania

5.9 inc/stos.hh File Reference

plik zawiera definicje klasy stos

```
#include "tablica.hh"
#include "instos.hh"
```

Include dependency graph for stos.hh:



Classes

- class [stos](#)

Klasa stos - jeden z abstrakcyjnych typow danych.

5.9.1 Detailed Description

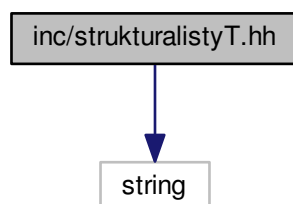
plik zawiera definicje klasy stos

5.10 inc/strukturalistyT.hh File Reference

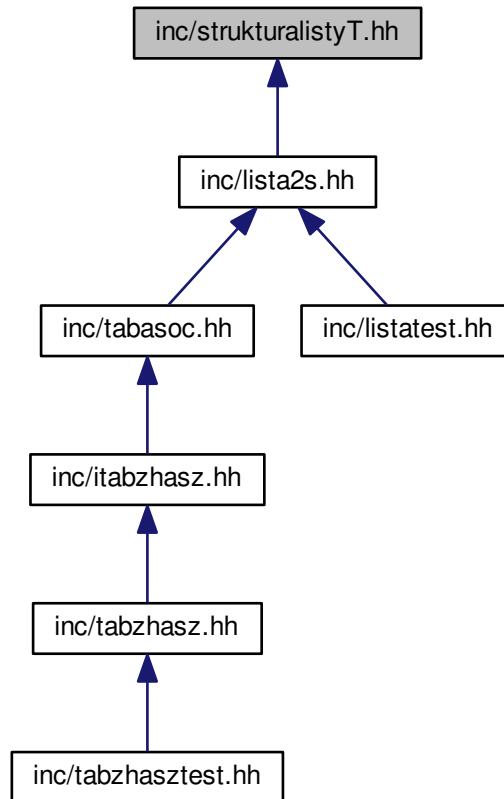
plik zawiera strukture potrzebna do dzialania listy

```
#include <string>
```

Include dependency graph for strukturalistyT.hh:



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



Classes

- struct [StrListyT< typ >](#)

5.10.1 Detailed Description

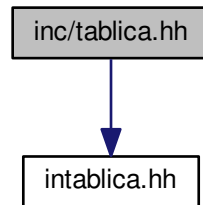
plik zawiera strukture potrzebna do dzialania listy

5.11 inc/tablica.hh File Reference

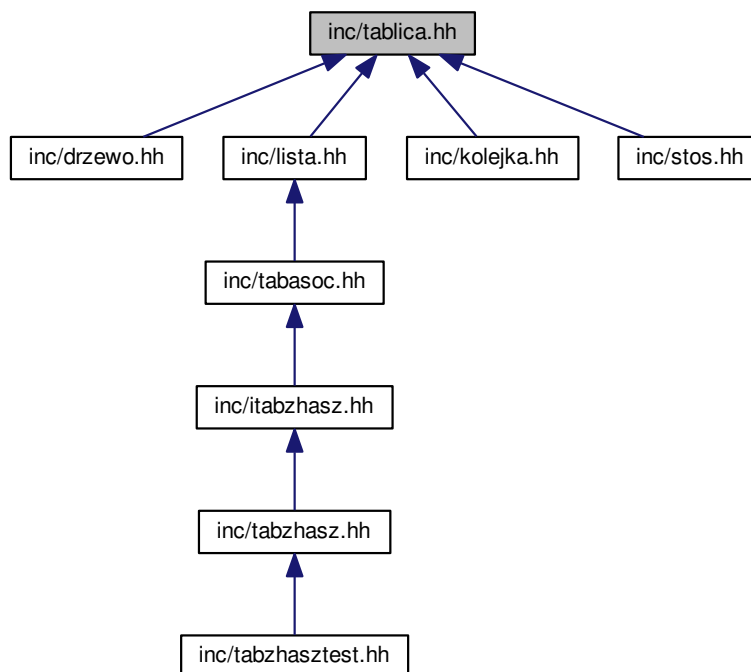
plik zawiera klase tablica

```
#include "intablica.hh"
```

Include dependency graph for tablica.hh:



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



Classes

- class `tablica< typ >`
Klasa tablica Klasa ma w swoim skladzie metody sluzace do zarzadzania tablica dynamiczna uzywane przez inne typy danych, moze byc tez uzywana samodzielnie.
- class `tablica< typ >::empty`
obiekt zwracany podczas wystapienia wyjatku wystepujacego gdy nastapi proba odzialywania na dane ktorych kontener nie posiada

- class [tablica< typ >::bad_index](#)

obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi odwołanie do danych o złym indeksie

5.11.1 Detailed Description

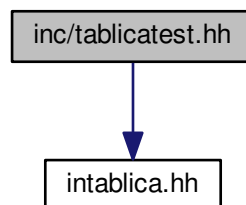
plik zawiera klase tablica

5.12 inc/tablicatest.hh File Reference

plik zawiera klase tablicatest

```
#include "intablica.hh"
```

Include dependency graph for tablicatest.hh:



Classes

- class [tablicatest](#)
- class [tablicatest::empty](#)

metoda sprawdza czy kontener jest pusty

- class [tablicatest::bad_index](#)

obiekt zwracany podczas wystąpienia wyjątku występującego gdy nastąpi próba odzyskiwania na dane których kontener nie posiada

5.12.1 Detailed Description

plik zawiera klase tablicatest

Index

countLines
Dictionary, 8

Dictionary, 8
countLines, 8
getWords, 8
RandomNumber, 8
RandomWords, 8
setWords, 9

Drzewo, 9

getWords
Dictionary, 8

iDrzewo< typ >, 11
iRunnable, 15
ITabAsoc, 15
ITabZHasz, 16
inKolejka, 12
inLista< typ >, 12
inStos, 13
inTablica< typ >, 14
inc/inlista.hh, 31
inc/intablica.hh, 32
inc/irunnable.hh, 33
inc/kolejka.hh, 33
inc/lista.hh, 34
inc/lista2s.hh, 36
inc/listatest.hh, 37
inc/sort.hh, 38
inc/stos.hh, 38
inc/strukturalistyT.hh, 39
inc/tablica.hh, 40
inc/tablicatest.hh, 42

kolejka, 17

lista< typ >, 18
lista< typ >::bad_index, 7
lista< typ >::empty, 10
lista2s< typ >, 19
lista2s< typ >::bad_index, 7
lista2s< typ >::empty, 10
listatest, 21

RandomNumber
Dictionary, 8
RandomWords
Dictionary, 8

setWords

Dictionary, 9
stoper, 22
stos, 22
StrListyT< typ >, 24

TabAsoc, 24
TabZHasz, 29
TabZHaszTest, 30
tablica< typ >, 25
tablica< typ >::bad_index, 7
tablica< typ >::empty, 10
tablicatest, 27
tablicatest::bad_index, 8
tablicatest::empty, 10