

Pomiar czasu dynamicznej alokacji pamięci

Generated by Doxygen 1.8.6

Sun Mar 13 2016 20:47:33

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	IRunnable Class Reference	7
4.1.1	Detailed Description	7
4.1.2	Member Function Documentation	7
4.1.2.1	prepare	7
4.1.2.2	run	8
4.2	IStoper Class Reference	8
4.2.1	Detailed Description	8
4.2.2	Member Function Documentation	8
4.2.2.1	dumpToFile	8
4.2.2.2	getElapsedTime	8
4.2.2.3	start	9
4.2.2.4	stop	9
4.3	Sedzia Class Reference	9
4.3.1	Detailed Description	9
4.3.2	Member Function Documentation	9
4.3.2.1	setOff	9
4.4	Stoper Class Reference	10
4.4.1	Detailed Description	12
4.4.2	Member Function Documentation	12
4.4.2.1	dumpToFile	12
4.4.2.2	getElapsedTime	12
4.4.2.3	start	13

4.4.2.4	stop	13
4.4.3	Member Data Documentation	14
4.4.3.1	_start	14
4.4.3.2	_stop	14
4.4.3.3	my_file	14
4.5	Tablica Class Reference	14
4.5.1	Detailed Description	15
4.5.2	Constructor & Destructor Documentation	16
4.5.2.1	Tablica	16
4.5.2.2	~Tablica	16
4.5.3	Member Function Documentation	16
4.5.3.1	getSize	16
4.5.3.2	increaseSize	16
4.5.3.3	isFull	17
4.5.3.4	prepare	17
4.5.3.5	run	17
4.5.4	Member Data Documentation	18
4.5.4.1	current_size	18
4.5.4.2	desired_size	18
4.5.4.3	elements	18
4.5.4.4	index	18
5	File Documentation	21
5.1	/home/kkuczaj/PAMSI/lab_07.03/prj/inc/IRunnable.h File Reference	21
5.1.1	Detailed Description	21
5.2	/home/kkuczaj/PAMSI/lab_07.03/prj/inc/IStoper.h File Reference	22
5.2.1	Detailed Description	22
5.3	/home/kkuczaj/PAMSI/lab_07.03/prj/inc/Sedzia.h File Reference	22
5.3.1	Detailed Description	23
5.4	/home/kkuczaj/PAMSI/lab_07.03/prj/inc/Stoper.h File Reference	23
5.4.1	Detailed Description	24
5.5	/home/kkuczaj/PAMSI/lab_07.03/prj/inc/Tablica.h File Reference	25
5.5.1	Detailed Description	25
5.6	/home/kkuczaj/PAMSI/lab_07.03/prj/src/main.cpp File Reference	26
5.6.1	Detailed Description	26
5.6.2	LICENCJA	26
5.6.3	OPIS	26
5.6.4	Function Documentation	27
5.6.4.1	main	27
5.7	/home/kkuczaj/PAMSI/lab_07.03/prj/src/Sedzia.cpp File Reference	27

5.8	/home/kkuczaj/PAMSI/lab_07.03/prj/src/Stoper.cpp File Reference	28
5.9	/home/kkuczaj/PAMSI/lab_07.03/prj/src/Tablica.cpp File Reference	28

Chapter 1

Opis programu

Author

Kamil Kuczaj 218478@student.pwr.edu.pl

1.1 Wstęp

Program został zbudowany modułowo. W folderze `inc/` znajdują się pliki nagłówkowe. Folder `src/` zawiera pliki źródłowe. W głównym folderze zbudowany został Makefile. Pliki obiektowe są budowane w folderze `obj/` a następnie linkowane do głównego folderu (`prj/`). Testowano przy wykorzystaniu kompilatora `g++` w wersji 4.8.4 na systemie Linux Ubuntu 14.04.04 opartego o jądro 4.2.0-30-generic.

1.2 Licencja

Program udostępniam na licencji GPLv3.

1.3 Instalacja

Aby zbudować i jednocześnie odpalić program: `$ make`

Aby pozbyć się plików z końcówką `*~` lub zaczynających się na `#*`: `$ make order`

Aby pozbyć się programu wykonywalnego oraz plików obiektowych: `$ make clean`

Aby wyświetlić pomoc do pliku Makefile: `$ make help`

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

IRunnable	7
Tablica	14
IStoper	8
Stoper	10
Sedzia	9

Chapter 3

Class Index

3.1 Class List

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

IRunnable	Interfejs dla biegacza	7
IStoper	Interfejs dla stopera	8
Sedzia	Implementacja klasy Sedzia	9
Stoper	Implementacja klasy Stoper	10
Tablica	Klasa tablica, w której odbywa się zapis dynamiczny elementów typu int	14

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

/home/kkuczaj/PAMSI/lab_07.03/prj/inc/ IRunnable.h	
Naglowek zawierajacy interfejs dla biegacza	21
/home/kkuczaj/PAMSI/lab_07.03/prj/inc/ IStoper.h	
Naglowek zawierajacy interfejs dla stopera	22
/home/kkuczaj/PAMSI/lab_07.03/prj/inc/ Sedzia.h	
Naglowek opisujacy implementacje Sedziego	22
/home/kkuczaj/PAMSI/lab_07.03/prj/inc/ Stoper.h	
Implementacja interfejsu IStoper w klasie Stoper	23
/home/kkuczaj/PAMSI/lab_07.03/prj/inc/ Tablica.h	
Implementacja interfejsu IRunnable	25
/home/kkuczaj/PAMSI/lab_07.03/prj/src/ main.cpp	26
/home/kkuczaj/PAMSI/lab_07.03/prj/src/ Sedzia.cpp	27
/home/kkuczaj/PAMSI/lab_07.03/prj/src/ Stoper.cpp	28
/home/kkuczaj/PAMSI/lab_07.03/prj/src/ Tablica.cpp	28

Chapter 5

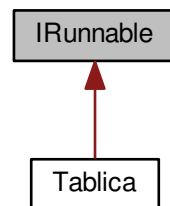
Class Documentation

5.1 IRunnable Class Reference

Interfejs dla biegacza.

```
#include <IRunnable.h>
```

Inheritance diagram for IRunnable:



Private Member Functions

- virtual void `prepare` (unsigned int size)=0
- virtual void `run` ()=0

5.1.1 Detailed Description

Interfejs dla biegacza.

Klasa abstrakcyjna z metodami czysto wirtualnymi.

5.1.2 Member Function Documentation

5.1.2.1 virtual void IRunnable::prepare (unsigned int *size*) [private],[pure virtual]

Implemented in [Tablica](#).

5.1.2.2 `virtual void IRunnable::run () [private],[pure virtual]`

Implemented in [Tablica](#).

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

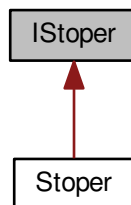
- [/home/kkuczaj/PAMSI/lab_07.03/prj/inc/IRunnable.h](#)

5.2 IStoper Class Reference

Interfejs dla stopera.

```
#include <IStoper.h>
```

Inheritance diagram for IStoper:



Public Member Functions

- `virtual void start ()=0`
- `virtual void stop ()=0`
- `virtual double getElapsedTime ()=0`
- `virtual void dumpToFile ()=0`

5.2.1 Detailed Description

Interfejs dla stopera.

Klasa abstrakcyjna z metodami czysto wirtualnymi.

5.2.2 Member Function Documentation

5.2.2.1 `virtual void IStoper::dumpToFile () [pure virtual]`

Implemented in [Stoper](#).

5.2.2.2 `virtual double IStoper::getElapsedTime () [pure virtual]`

Implemented in [Stoper](#).

5.2.2.3 `virtual void IStoper::start () [pure virtual]`

Implemented in [Stoper](#).

5.2.2.4 `virtual void IStoper::stop () [pure virtual]`

Implemented in [Stoper](#).

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

- `/home/kkuczaj/PAMSI/lab_07.03/prj/inc/IStoper.h`

5.3 Sedzia Class Reference

Implementacja klasy [Sedzia](#).

```
#include <Sedzia.h>
```

Public Member Functions

- `bool setOff (unsigned int how_many)`

Funkcja, w ktorej odbywa sie bieg.

5.3.1 Detailed Description

Implementacja klasy [Sedzia](#).

[Sedzia](#) wykorzystuje elementy klasy [Stoper](#) oraz klasy [Tablica](#). Mierzy czas wypelniania elemntow Tablicy.

5.3.2 Member Function Documentation

5.3.2.1 `bool Sedzia::setOff (unsigned int how_many)`

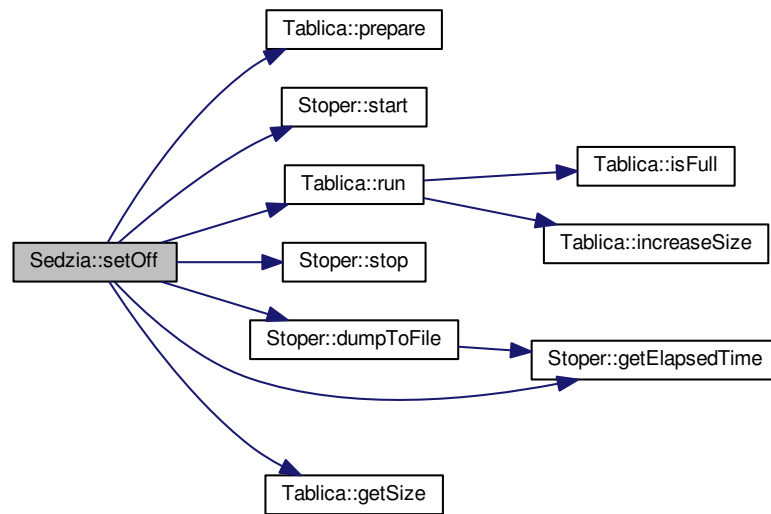
Funkcja, w ktorej odbywa sie bieg.

Podczas wykonywania tej funkcji uruchamiany jest [Stoper](#) oraz wypelniany jest element klasy Tablica po uprzednim jej przygotowaniu.

Parameters

<i>how_many</i>	Informacja iloma elementami ma zostac wypelniona tablica.
-----------------	---

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:

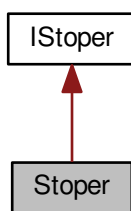
- `/home/kkuczaj/PAMSI/lab_07.03/prj/inc/Sedzia.h`
- `/home/kkuczaj/PAMSI/lab_07.03/prj/src/Sedzia.cpp`

5.4 Stoper Class Reference

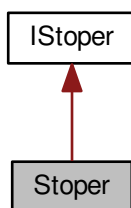
Implementacja klasy [Stoper](#).

```
#include <Stoper.h>
```

Inheritance diagram for Stoper:



Collaboration diagram for Stoper:



Public Member Functions

- virtual void [start](#) ()
Implementacja funkcji [start\(\)](#) z interfejsu [IStoper](#).
- virtual void [stop](#) ()
Implementacja funkcji [stop\(\)](#) z interfejsu [IStoper](#).
- virtual double [getElapsedTime](#) ()
Implementacja funkcji [getElapse\(\)](#) z interfejsu [IStoper](#).
- virtual void [dumpToFile](#) ()
Implementacja funkcji [dumpToFile\(\)](#) z interfejsu [IStoper](#).

Private Attributes

- clock_t [_start](#)
Moment startu stopera.
- clock_t [_stop](#)
Moment zatrzymania stopera.
- std::fstream [my_file](#)
Strumień zapisu do pliku.

Additional Inherited Members

5.4.1 Detailed Description

Implementacja klasy [Stoper](#).

W klasie [Stoper](#) zostały zaimplementowane metody pozwalające na pomiar czasu. Pomiar czasu odbywa się dzięki bibliotece `<ctime>` a zapis do pliku korzysta z biblioteki `<fstream>`.

5.4.2 Member Function Documentation

5.4.2.1 `void Stoper::dumpToFile ()` [virtual]

Implementacja funkcji [dumpToFile\(\)](#) z interfejsu [IStoper](#).

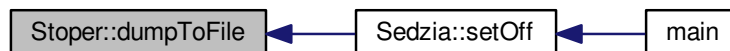
Zapisuje zmierzony czas do pliku o nazwie "tmp_czasy.txt".

Implements [IStoper](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.4.2.2 `double Stoper::getElapsedTime ()` [virtual]

Implementacja funkcji `getElapse()` z interfejsu [IStoper](#).

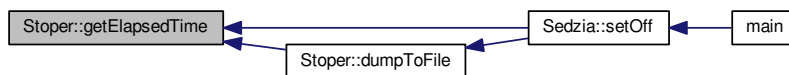
Oblicza czas pomiędzy czasem zapisanym w zmiennych `_start` i `_stop`.

Returns

Zwraca zmierzony czas - roznica pomiedzy polem `_start` a polem `_stop`.

Implements [IStoper](#).

Here is the caller graph for this function:



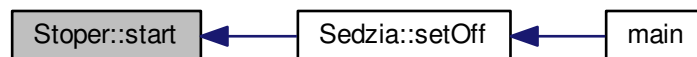
5.4.2.3 void Stoper::start () [virtual]

Implementacja funkcji [start\(\)](#) z interfejsu [IStoper](#).

Zapisuje moment uruchomienia stopera.

Implements [IStoper](#).

Here is the caller graph for this function:



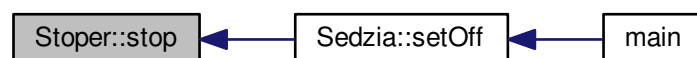
5.4.2.4 void Stoper::stop () [virtual]

Implementacja funkcji [stop\(\)](#) z interfejsu [IStoper](#).

Zapisuje moment zatrzymania stopera.

Implements [IStoper](#).

Here is the caller graph for this function:



5.4.3 Member Data Documentation

5.4.3.1 `clock_t Stoper::_start` [private]

Moment startu stopera.

Element przechowujący informacje o czasie systemowym w momencie uruchomienia stopera. Element typu `clock_t`.

5.4.3.2 `clock_t Stoper::_stop` [private]

Moment zatrzymania stopera.

Element przechowujący informacje o czasie systemowym w momencie zatrzymania stopera. Element typu `clock_t`.

5.4.3.3 `std::fstream Stoper::my_file` [private]

Strumień zapisu do pliku.

Pole ułatwiające zapis do pliku.

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

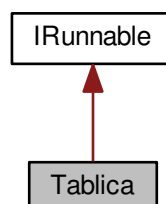
- [/home/kkuczaj/PAMSI/lab_07.03/prj/inc/Stoper.h](#)
- [/home/kkuczaj/PAMSI/lab_07.03/prj/src/Stoper.cpp](#)

5.5 Tablica Class Reference

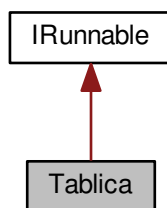
Klasa tablica, w której odbywa się zapis dynamiczny elementów typu `int`.

```
#include <Tablica.h>
```

Inheritance diagram for Tablica:



Collaboration diagram for Tablica:



Public Member Functions

- `Tablica` (int x=10)
Konstruktor parametryczny.
- `~Tablica` ()
Destruktor.
- virtual void `prepare` (unsigned int size)
Implementacja funkcji `prepare()` interfejsu `IRunnable`.
- virtual void `run` ()
Implementacja funkcji `run()` interfejsu `IRunnable`.
- unsigned int `getSize` ()
Zwraca aktualny rozmiar tablicy dynamicznej.

Private Member Functions

- bool `isFull` ()
Pozwala prosto okreslic, czy nalezy przydzielic pamiec.
- void `increaseSize` ()
Zwieksza rozmiar przydzielonej pamieci na stercie.

Private Attributes

- int * `elements`
Wskaźnik do początku tablicy dynamicznej.
- unsigned int `current_size`
Okresla aktualny rozmiar stosu.
- unsigned int `desired_size`
Okresla pozadany rozmiar stosu.
- unsigned int `index`
Okresla aktualny indeks.

5.5.1 Detailed Description

Klasa `tablica`, w której odbywa się zapis dynamiczny elementów typu `int`.

Implementuje metody interfejsu `IRunnable`. Zajmuje się dynamiczną alokacją pamięci.

5.5.2 Constructor & Destructor Documentation

5.5.2.1 `Tablica::Tablica (int x = 10)`

Konstruktor parametryczny.

Umożliwia określenie początkowego rozmiaru tablicy. W przypadku braku określenia tego rozmiaru przyjmuje domyślną wartość równą 10.

Parameters

<code>x</code>	Określa początkową wielkość przydzielonej pamięci. Domyślna wartość w przypadku braku podania to 10.
----------------	--

5.5.2.2 `Tablica::~~Tablica ()`

Destruktor.

Usuwa pamięć przypisaną komorce, na którą wskazuje pole `*elements`.

5.5.3 Member Function Documentation

5.5.3.1 `unsigned int Tablica::getSize ()`

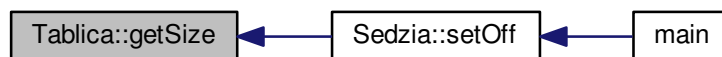
Zwraca aktualny rozmiar tablicy dynamicznej.

Zwraca wartość pola `current_size`.

Returns

Zwraca wartość typu `unsigned int`, gdyż takiego typu jest zmienna `current_size`.

Here is the caller graph for this function:

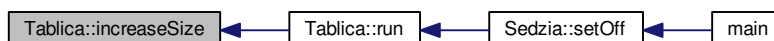


5.5.3.2 `void Tablica::increaseSize ()` `[private]`

Zwiększa rozmiar przydzielonej pamięci na stacku.

Metoda prywatna. Kopiuje elementy starej pamięci do komórki z nowo-przydzieloną pamięcią. Usuwa starą pamięć.

Here is the caller graph for this function:



5.5.3.3 bool Tablica::isFull () [private]

Pozwala prosto okreslic, czy nalezy przydzielic pamiec.

Metoda prywatna. Sluzy do okreslania czy nalezy wywolac metode [increaseSize\(\)](#).

Returns

true Pamiec pelna. Nalezy zwiekszyc rozmiar.
false Jest jeszcze wolne miejsce.

Here is the caller graph for this function:



5.5.3.4 void Tablica::prepare (unsigned int size) [virtual]

Implementacja funkcji [prepare\(\)](#) interfesju [IRunnable](#).

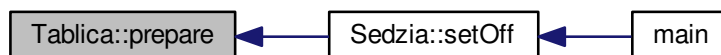
Zapisuje pozadany rozmiar do pola desired_size.

Parameters

<i>size</i>	Parametr typu unsigned int, gdyz rozmiar nie powinien nigdy byc ujemny. Jego wartosc zapisywana jest do pola desired_size.
-------------	--

Implements [IRunnable](#).

Here is the caller graph for this function:



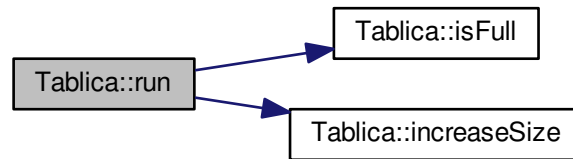
5.5.3.5 void Tablica::run () [virtual]

Implementacja funkcji [run\(\)](#) interfesju [IRunnable](#).

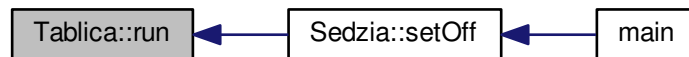
Uruchamia "bieg", w ktorym nastepuje zapis elementow do poszczegolnych elementow tablicy dynamicznej. Tam odbywa sie alokacja pamieci oraz instrukcje warunkowe.

Implements [IRunnable](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.5.4 Member Data Documentation

5.5.4.1 `unsigned int Tablica::current_size` [private]

Okresla aktualny rozmiar stosu.

Pole prywatne typu `unsigned int`, gdyz rozmiar nigdy nie powinien byc ujemny.

5.5.4.2 `unsigned int Tablica::desired_size` [private]

Okresla pozadany rozmiar stosu.

Pole prywatne typu `unsigned int`, gdyz rozmiar nigdy nie powinien byc ujemny. Zadawane w funkcji [prepare\(\)](#).

5.5.4.3 `int* Tablica::elements` [private]

Wskaźnik do początku tablicy dynamicznej.

Wskazuje na adres w pamięci serty. Pole prywatne.

5.5.4.4 `unsigned int Tablica::index` [private]

Okresla aktualny indeks.

Pole prywatne typu `unsigned int`, gdyz indeks nigdy nie powinien byc ujemny. Przechowuje indeks, pierwszej wolnej komórki pamięci, do ktorego mozliwy bedzie zapis.

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

- [/home/kkuczaj/PAMSI/lab_07.03/prj/inc/Tablica.h](#)
- [/home/kkuczaj/PAMSI/lab_07.03/prj/src/Tablica.cpp](#)

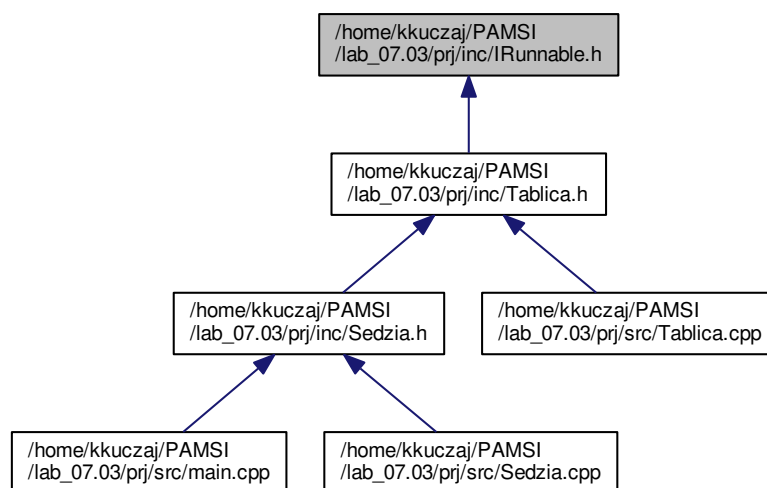
Chapter 6

File Documentation

6.1 /home/kkuczaj/PAMSI/lab_07.03/prj/inc/IRunnable.h File Reference

Naglowek zawierajacy interfejs dla biegacza.

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



Classes

- class [IRunnable](#)
Interfejs dla biegacza.

6.1.1 Detailed Description

Naglowek zawierajacy interfejs dla biegacza.

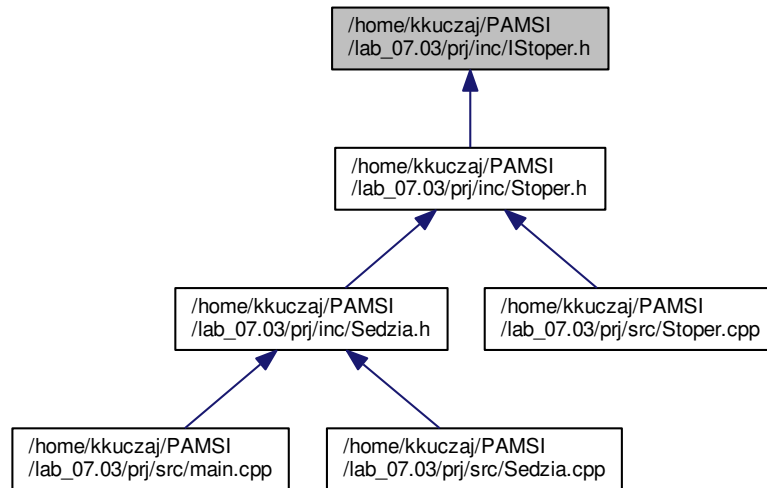
Author

Kamil Kuczaj

6.2 /home/kkuczaj/PAMSI/lab_07.03/prj/inc/IStoper.h File Reference

Naglowek zawierajacy interfejs dla stopera.

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



Classes

- class [IStoper](#)

Interfejs dla stopera.

6.2.1 Detailed Description

Naglowek zawierajacy interfejs dla stopera.

Author

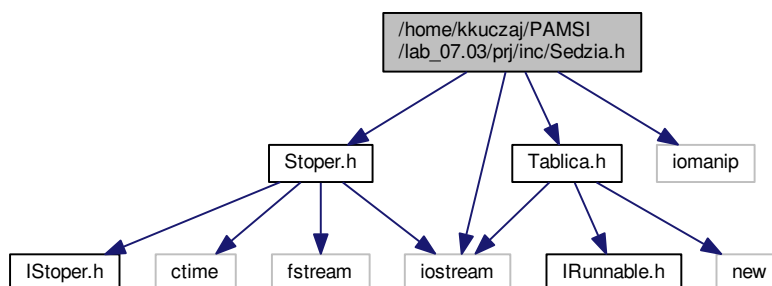
Kamil Kuczaj

6.3 /home/kkuczaj/PAMSI/lab_07.03/prj/inc/Sedzia.h File Reference

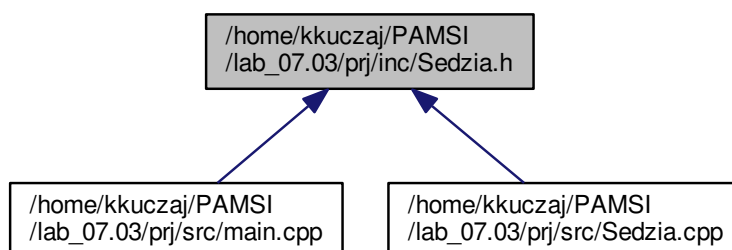
Naglowek opisujacy implementacje Sedziego.

```
#include "Stoper.h"
#include "Tablica.h"
#include <iostream>
#include <iomanip>
```

Include dependency graph for Sedzia.h:



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



Classes

- class [Sedzia](#)

Implementacja klasy [Sedzia](#).

6.3.1 Detailed Description

Nagłówek opisujący implementację Sedziego.

Author

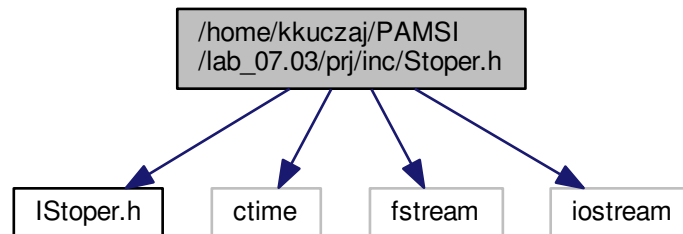
Kamil Kuczaj

6.4 /home/kkuczaj/PAMSI/lab_07.03/prj/inc/Stoper.h File Reference

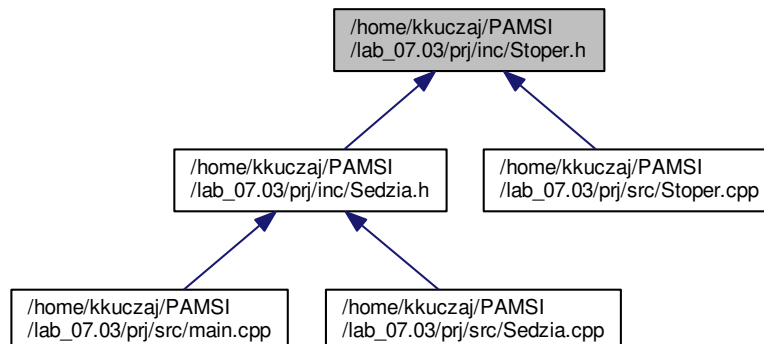
Implementacja interfejsu [IStoper](#) w klasie [Stoper](#).

```
#include "IStoper.h"
#include <ctime>
#include <fstream>
#include <iostream>
```

Include dependency graph for Stoper.h:



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



Classes

- class [Stoper](#)

Implementacja klasy [Stoper](#).

6.4.1 Detailed Description

Implementacja interfejsu [IStoper](#) w klasie [Stoper](#).

Author

Kamil Kuczaj

6.5 /home/kkuczaj/PAMSI/lab_07.03/prj/inc/Tablica.h File Reference

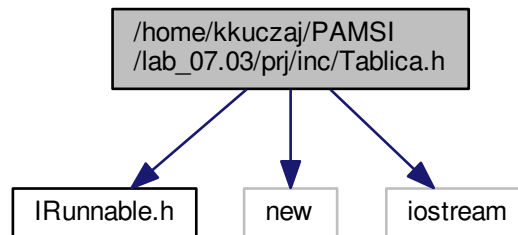
Implementacja interfejsu [IRunnable](#).

```
#include "IRunnable.h"
```

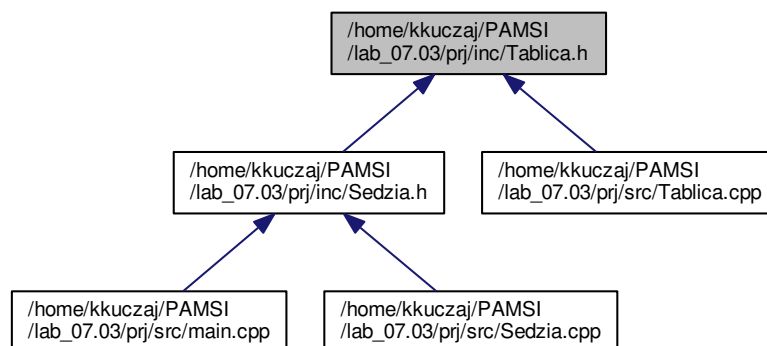
```
#include <new>
```

```
#include <iostream>
```

Include dependency graph for Tablica.h:



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



Classes

- class [Tablica](#)

Klasa tablica, w której odbywa się zapis dynamiczny elementów typu int.

6.5.1 Detailed Description

Implementacja interfejsu [IRunnable](#).

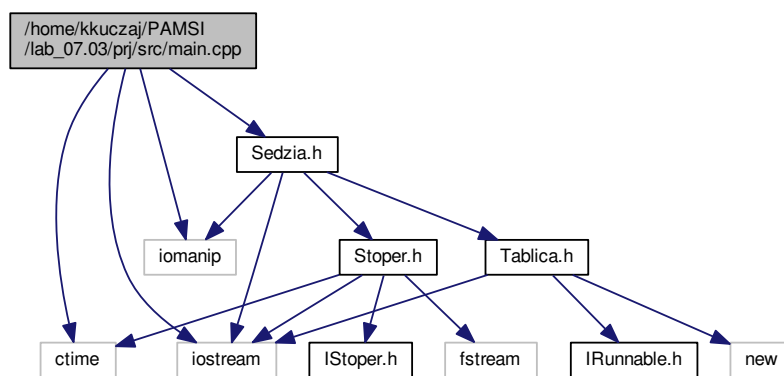
Author

Kamil Kuczaj

6.6 /home/kkuczaj/PAMSI/lab_07.03/prj/src/main.cpp File Reference

```
#include <iostream>
#include <ctime>
#include <iomanip>
#include "Sedzia.h"
```

Include dependency graph for main.cpp:



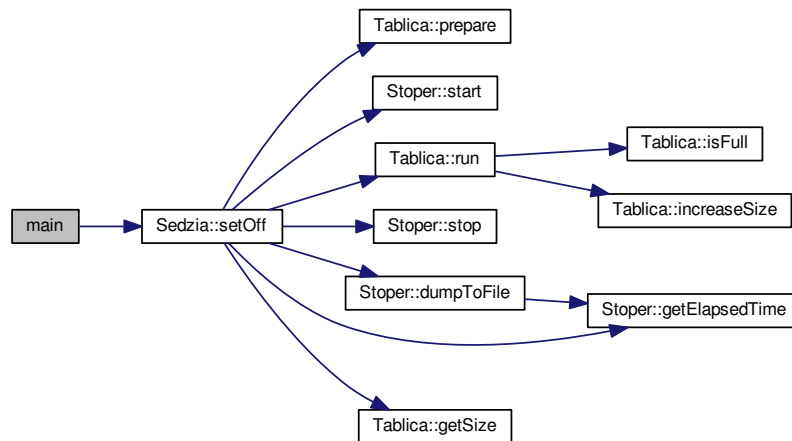
Functions

- int [main](#) (int argc, char **argv)

6.6.1 Function Documentation

6.6.1.1 `int main (int argc, char ** argv)`

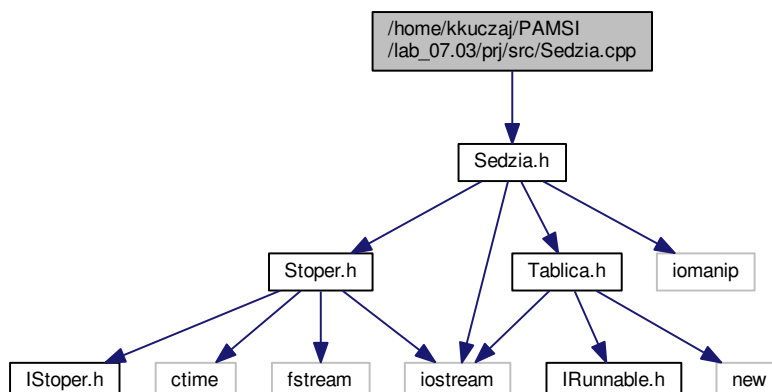
Here is the call graph for this function:



6.7 /home/kkuczaj/PAMSI/lab_07.03/prj/src/Sedzia.cpp File Reference

```
#include "Sedzia.h"
```

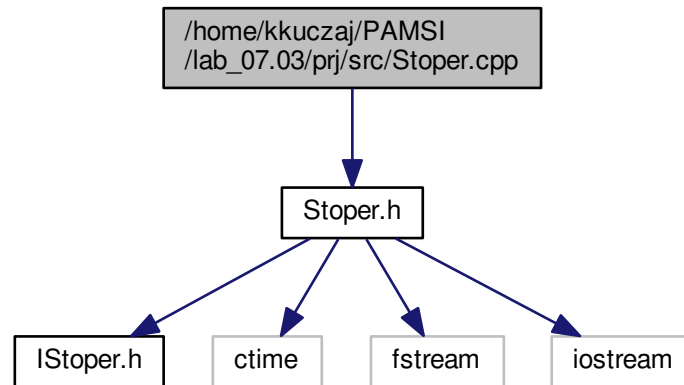
Include dependency graph for Sedzia.cpp:



6.8 /home/kkuczaj/PAMSI/lab_07.03/prj/src/Stoper.cpp File Reference

```
#include "Stoper.h"
```

Include dependency graph for Stoper.cpp:



6.9 /home/kkuczaj/PAMSI/lab_07.03/prj/src/Tablica.cpp File Reference

```
#include "Tablica.h"
```

Include dependency graph for Tablica.cpp:

