Spedycja

Generated by Doxygen 1.8.14

Contents

Chapter 1

Class Index

1.1 Class List

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

droga . . . . . ??
miasto . . . . . ??

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

ConsoleApplication1.cpp . . . . . ??
funkcje.cpp . . . . . ??
funkcje.h . . . . . ??
pch.cpp . . . . . ??
pch.h . . . . . ??
struktury.cpp . . . . . ??
struktury.h . . . . . ??

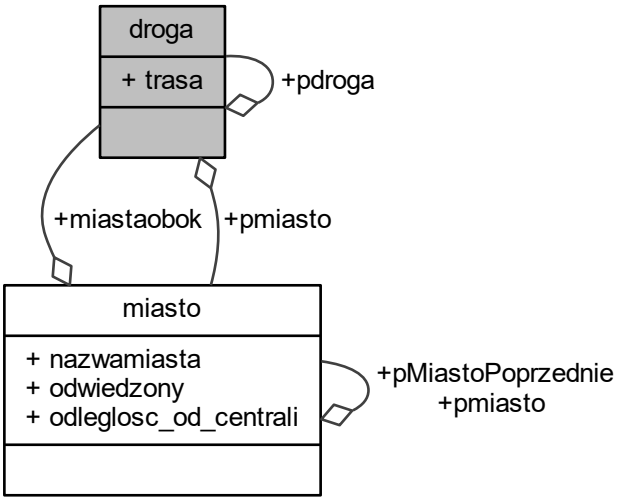
Chapter 3

Class Documentation

3.1 droga Struct Reference

`#include <struktury.h>`

Collaboration diagram for droga:



Public Attributes

- int trasa
- droga * pdroga
- miasto * pmiasto

3.1.1 Detailed Description

Struktura droga

Parameters

trasa	odleglosc miedzy miastami
pdroga	wskaznik na nastepna droge
pmiasto	wskaznik na odpowiednie miasto

3.1.2 Member Data Documentation

3.1.2.1 pdroga

```
droga* droga::pdroga
```

3.1.2.2 pmiasto

```
miasto* droga::pmiasto
```

3.1.2.3 trasa

```
int droga::trasa
```

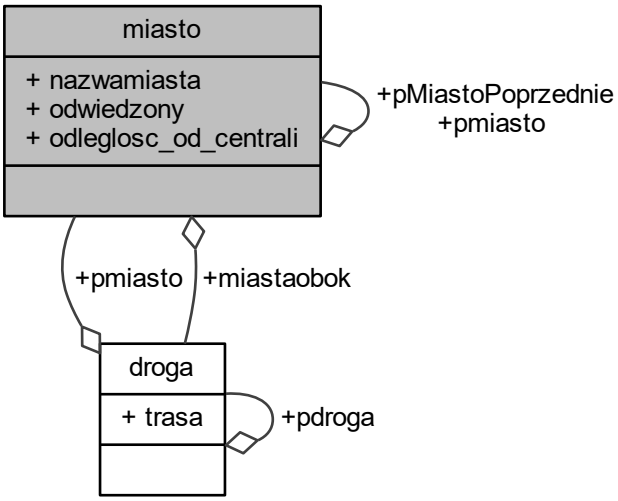
The documentation for this struct was generated from the following file:

- struktury.h

3.2 miasto Struct Reference

```
#include <struktury.h>
```

Collaboration diagram for miasto:



Public Attributes

- std::string nazwamiasta
- miasto * pmiasto
- droga * miastaobok
- bool odwiedzony
- int odleglosc_od_centrali
- miasto * pMiastoPoprzednie

3.2.1 Detailed Description

Struktura miasto

Parameters

nazwamiasta	nazwa miasta
pmiasto	wskaznik na nastepne miasto
miastaobok	wskaznik na pierwszy element list drog
odwiedzony	zwraca true, gdy miasto zostalo odwiedzone i false i nie zopstalo odwiedzone
odleglosc_od_centrali	odleglosc od centrali
pMiastoPoprzednie	wskaznik na miasto poprzednie

3.2.2 Member Data Documentation

3.2.2.1 miastaobok

```
droga* miasto::miastaobok
```

3.2.2.2 nazwamiasta

```
std::string miasto::nazwamiasta
```

3.2.2.3 odleglosc_od_centrali

```
int miasto::odleglosc_od_centrali
```

3.2.2.4 odwiedzony

```
bool miasto::odwiedzony
```

3.2.2.5 pmiasto

```
miasto* miasto::pmiasto
```

3.2.2.6 pMiastoPoprzednie

```
miasto* miasto::pMiastoPoprzednie
```

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

- struktury.h

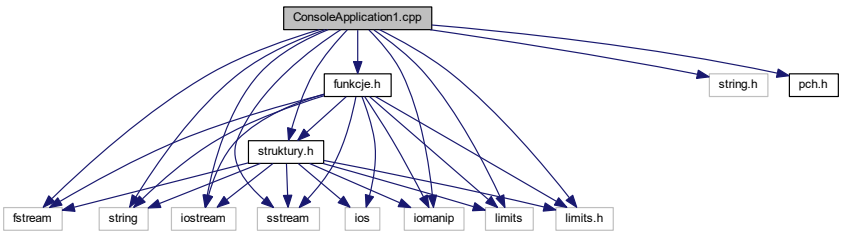
Chapter 4

File Documentation

4.1 ConsoleApplication1.cpp File Reference

```
#include <iostream>
#include <fstream>
#include <string>
#include <sstream>
#include <iomanip>
#include <limits>
#include <limits.h>
#include "string.h"
#include "pch.h"
#include "struktury.h"
#include "funkcje.h"
```

Include dependency graph for ConsoleApplication1.cpp:



Macros

- #define debug(x) std::cerr << "(" << __LINE__ << ") " << #x << " == " << (x) << std::endl;

Functions

- int main (int ile, char **params)

4.1.1 Macro Definition Documentation

4.1.1.1 debug

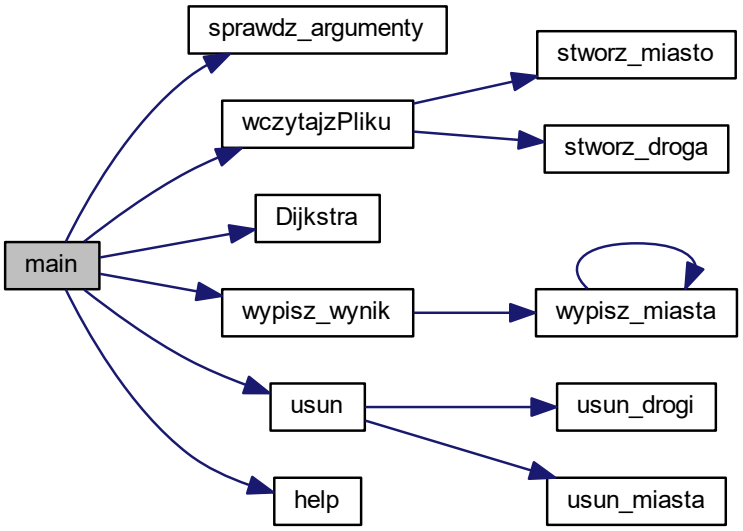
```
#define debug(
             x ) std::cerr << "(" << __LINE__ << ") " << #x << " == " << (x) << std↩
::endl;
```

4.1.2 Function Documentation

4.1.2.1 main()

```
int main (
             int ile,
             char ** params )
```

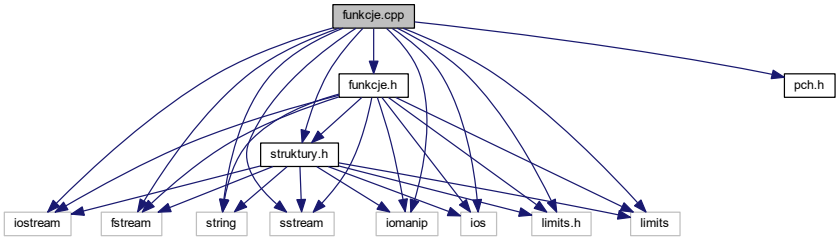
Here is the call graph for this function:



4.2 funkcje.cpp File Reference

```
#include <iostream>
#include <fstream>
#include <string>
#include <sstream>
#include <limits>
#include <iomanip>
#include <ios>
#include <limits.h>
#include "pch.h"
#include "struktury.h"
#include "funkcje.h"
```

Include dependency graph for funkcje.cpp:



Functions

- miasto * stworz_miasto (miasto *&pHead_miasto, const std::string &nowanazwa)
- void stworz_droga (miasto *&pHead_miasto, int kilometry, miasto *&nowe_miasto1, miasto *&nowe_miasto2)
- void wypisz_droga (droga *pHead_droga)
- void wypisz_miasto (miasto *pHead)
- bool Dijkstra (const std::string &startowy, miasto *&pHead)
- void wypisz_miasta (miasto *pHead, std::ostream &wyjscie)
- void wypisz_wynik (miasto *pHead, const std::string &wyjscie)
- bool sprawdz_argumenty (int ile, char **params, std::string &wejscie, std::string &wyjscie, std::string &start)
- void wczytajzPliku (const std::string &wejscie, miasto *&pGlowa)
- void usun_drogi (miasto *pmiasto)
- void usun_miasta (miasto *&pHead)
- void usun (miasto *glowa_miasta)
- void help (int ile, char **params)

4.2.1 Function Documentation

4.2.1.1 Dijkstra()

```
bool Dijkstra (
            const std::string & startowy,
            miasto *& pHead )
```

Funkcja, wykorzystujca algorytm Dijkstry do znalezienia najkrotszych drog z miasta startowego do pozostalych miast

Parameters

startowy	miasto, od ktorego beda rozpoczynac sie wszystkie trasy
pHead	wskaznik na pierwszy element listy

Returns

Funkcja zwraca true, gdy miasto startowe bylo w liscie i false, gdy nie bylo w liscie

Here is the caller graph for this function:

4.2.1.2 help()

```
void help (
            int ile,
            char ** params )
```

Funkcja wyswietla pomoc

Parameters

ile	ilosc argumentow
params	tablica argumentow

Returns

Funkcja nie zwraca niczego.

Here is the caller graph for this function:

4.2.1.3 sprawdz_argumenty()

```
bool sprawdz_argumenty (
            int ile,
            char ** params,
            std::string & wejscie,
            std::string & wyjscie,
            std::string & start )
```

Funkcja sprawdza argumenty wywolania programu

Parameters

	ile	ilosc argumentow
	params	tablica argumentow
out	*wejscie*	plik wejsciowy
out	*wyjscie*	plik wyjsciowy
out	*start*	miasto startowe

Returns

Funkcja zwraca true, gdy podane argumenty byly poprawne i false, gdy byly bledne

Here is the caller graph for this function:

4.2.1.4 stworz_droga()

```
void stworz_droga (
            miasto *& pHead_miasto,
            int kilometry,
            miasto *& nowe_miasto1,
            miasto *& nowe_miasto2 )
```

Funkcja dodaje nowe elementy do listy drParameters *pHead_miasto* wskaznik na pierwszy element listy

kilometry odlegosc pomiedzy miastami

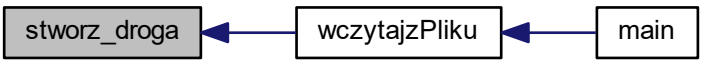
nowe_miasto1 wskaznik na miasto poczatkowe

nowe_miasto2 wskaznik na miasto docelowe

Returns

Funkcja nie zwraca niczego.

Here is the caller graph for this function:



4.2.1.5 stworz_miasto()

```
miasto* stworz_miasto (
            miasto *& pHead,
            const std::string & nowanazwa )
```

Funkcja dodaje nowe miasto do listy

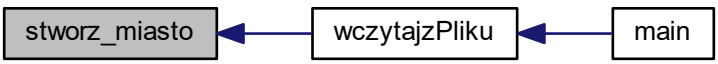
Parameters

pHead	wskaznik na pierwszy element listy
nowanazwa	nazwa miasta dodawanego do listy

Returns

Funkcja zwraca wskaznik na nowoutworzone miasto.

Here is the caller graph for this function:



4.2.1.6 usun()

```
void usun (
            miasto * glowa_miasta )
```

Funkcja usuwa liste miast i drog

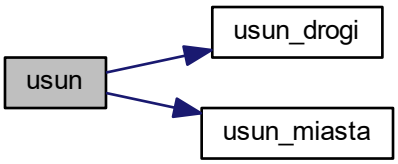
Parameters

glowa_miasta	wskaznik na pierwszy element listy miast

Returns

Funkcja nie zwraca niczego.

Here is the call graph for this function:



Here is the caller graph for this function:

4.2.1.7 usun_drogi()

```
void usun_drogi (
            miasto * pmiasto )
```

Funkcja usuwa listy drog wszystkich miast

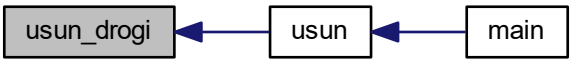
Parameters

pmiasto	wskaznik na kolejne miasto

Returns

Funkcja nie zwraca niczego.

Here is the caller graph for this function:



4.2.1.8 usun_miasta()

```
void usun_miasta (
            miasto *& pHead )
```

Funkcja usuwa liste miast

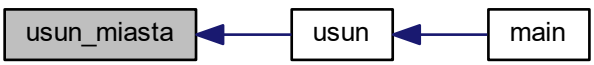
Parameters

pHead	wskaznik na pierwszy element listy

Returns

Funkcja nie zwraca niczego.

Here is the caller graph for this function:



4.2.1.9 wczytajzPliku()

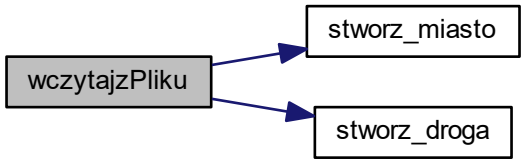
```
void wczytajzPliku (
            const std::string & wejscie,
            miasto *& pGlowa )
```

Funkcja wczytuje dane z pliku

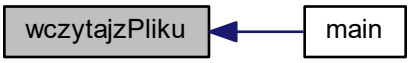
Parameters

wejscie	nazwa pliku wejsciowego
pGlowa	wskaznik na pierwszy element listy

Here is the call graph for this function:



Here is the caller graph for this function:



4.2.1.10 wypisz_droga()

```
void wypisz_droga (
            droga * pHead_droga )
```

Funkcja wypisuje liste drog

Parameters

pHead_droga	wskaznik na pierwszy element listy

Returns

Funkcja nie zwraca niczego.

Here is the caller graph for this function:

4.2.1.11 wypisz_miasta()

```
void wypisz_miasta (
            miasto * pHead,
            std::ostream & wyjscie )
```

Funkcja zapisuje trasy do pliku wyjsciowego

Parameters

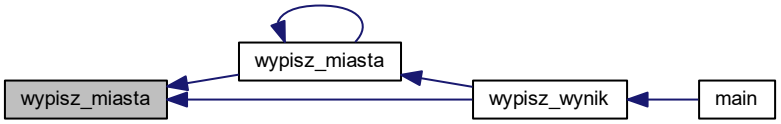
pHead	wskaznik na pierwszy element listy
wyjscie	nazwa pliku wyjsciowego

Returns

Funkcja nie zwraca niczego.

Here is the call graph for this function:

Here is the caller graph for this function:



4.2.1.12 wypisz_miasto()

```
void wypisz_miasto (
            miasto * pHead )
```

Funkcja wypisuje liste miast

Parameters

pHead	wskaznik na pierwszy element listy

Returns

Funkcja nie zwraca niczego.

Here is the call graph for this function:

4.2.1.13 wypisz_wynik()

```
void wypisz_wynik (
            miasto * pHead,
            const std::string & wyjscie )
```

Funkcja zapisuje wynik do pliku wyjsciowego

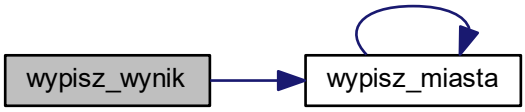
Parameters

in,out	*pHead*	wskaznik na pierwszy element listy
	wyjscie	nazwa pliku wyjsciowego

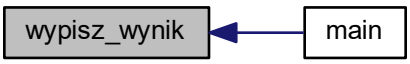
Returns

Funkcja nie zwraca niczego.

Here is the call graph for this function:



Here is the caller graph for this function:

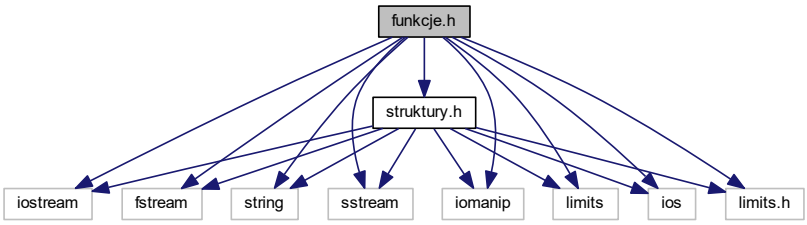


4.3 funkcje.h File Reference

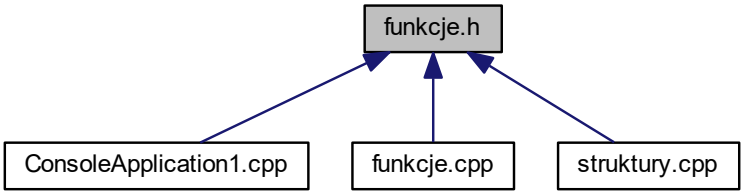
```
#include <iostream>
#include <fstream>
#include <string>
#include <sstream>
#include <iomanip>
#include <limits>
#include <ios>
#include <limits.h>
```

```
#include "struktury.h"
```

Include dependency graph for funkcje.h:



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



Functions

- miasto * stworz_miasto (miasto *&pHead, const std::string &nowanazwa)
- void stworz_droga (miasto *&pHead_miasto, int kilometry, miasto *&nowe_miasto1, miasto *&nowe_↩ miasto2)
- void wypisz_miasto (miasto *pHead)
- void wypisz_droga (droga *pHead_droga)
- bool Dijkstra (const std::string &startowy, miasto *&pHead)
- void wypisz_wynik (miasto *pHead, const std::string &wyjscie)
- void wypisz_miasta (miasto *pHead, std::ostream &wyjscie)
- bool sprawdz_argumenty (int ile, char **params, std::string &wejscie, std::string &wyjscie, std::string &start)
- void wczytajzPliku (const std::string &wejscie, miasto *&pGlowa)
- void usun_drogi (miasto *pmiasto)
- void usun_miasta (miasto *&pHead)
- void usun (miasto *glowa_miasta)
- void help (int ile, char **params)

4.3.1 Function Documentation

4.3.1.1 Dijkstra()

```
bool Dijkstra (
            const std::string & startowy,
            miasto *& pHead )
```

Funkcja, wykorzystujca algorytm Dijkstry do znalezienia najkrotszych drog z miasta startowego do pozostalych miast

Parameters

startowy	miasto, od ktorego beda rozpoczynac sie wszystkie trasy
pHead	wskaznik na pierwszy element listy

Returns

Funkcja zwraca true, gdy miasto startowe bylo w liscie i false, gdy nie bylo w liscie

Here is the caller graph for this function:

4.3.1.2 help()

```
void help (
            int ile,
            char ** params )
```

Funkcja wyswietla pomoc

Parameters

ile	ilosc argumentow
params	tablica argumentow

Returns

Funkcja nie zwraca niczego.

Here is the caller graph for this function:

4.3.1.3 sprawdz_argumenty()

```
bool sprawdz_argumenty (
            int ile,
            char ** params,
            std::string & wejscie,
            std::string & wyjscie,
            std::string & start )
```

Funkcja sprawdza argumenty wywolania programu

Parameters

	ile	ilosc argumentow
	params	tablica argumentow
out	*wejscie*	plik wejsciowy
out	*wyjscie*	plik wyjsciowy
out	*start*	miasto startowe

Returns

Funkcja zwraca true, gdy podane argumenty byly poprawne i false, gdy byly bledne

Here is the caller graph for this function:

4.3.1.4 stworz_droga()

```
void stworz_droga (
            miasto *& pHead_miasto,
            int kilometry,
            miasto *& nowe_miasto1,
            miasto *& nowe_miasto2 )
```

Funkcja dodaje nowe elementy do listy drParameters *pHead_miasto* wskaznik na pierwszy element listy

---

kilometry odlegosc pomiedzy miastami

---

nowe_miasto1 wskaznik na miasto poczatkowe

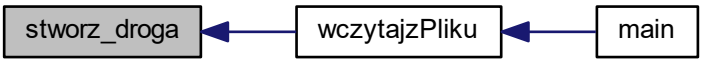
---

nowe_miasto2 wskaznik na miasto docelowe

Returns

Funkcja nie zwraca niczego.

Here is the caller graph for this function:



4.3.1.5 stworz_miasto()

```
miasto* stworz_miasto (
            miasto *& pHead,
            const std::string & nowanazwa )
```

Funkcja dodaje nowe miasto do listy

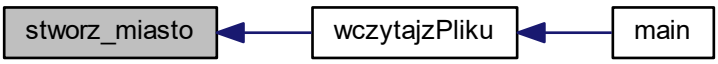
Parameters

pHead	wskaznik na pierwszy element listy
nowanazwa	nazwa miasta dodawanego do listy

Returns

Funkcja zwraca wskaznik na nowoutworzone miasto.

Here is the caller graph for this function:



4.3.1.6 usun()

```
void usun (
            miasto * glowa_miasta )
```

Funkcja usuwa liste miast i drog

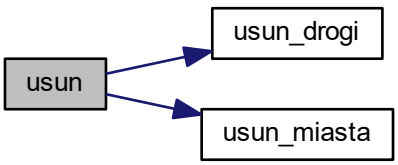
Parameters

glowa_miasta	wskaznik na pierwszy element listy miast

Returns

Funkcja nie zwraca niczego.

Here is the call graph for this function:



Here is the caller graph for this function:

4.3.1.7 usun_drogi()

```
void usun_drogi (
            miasto * pmiasto )
```

Funkcja usuwa listy drog wszystkich miast

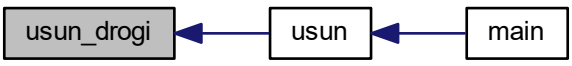
Parameters

pmiasto	wskaznik na kolejne miasto

Returns

Funkcja nie zwraca niczego.

Here is the caller graph for this function:



4.3.1.8 usun_miasta()

```
void usun_miasta (
            miasto *& pHead )
```

Funkcja usuwa liste miast

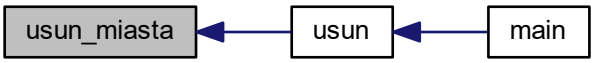
Parameters

pHead	wskaznik na pierwszy element listy

Returns

Funkcja nie zwraca niczego.

Here is the caller graph for this function:



4.3.1.9 wczytajzPliku()

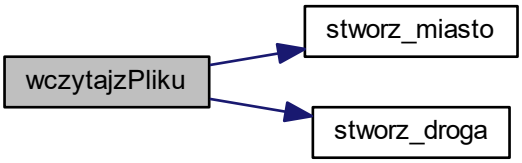
```
void wczytajzPliku (
            const std::string & wejscie,
            miasto *& pGlowa )
```

Funkcja wczytuje dane z pliku

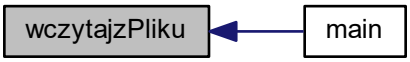
Parameters

wejscie	nazwa pliku wejsciowego
pGlowa	wskaznik na pierwszy element listy

Here is the call graph for this function:



Here is the caller graph for this function:



4.3.1.10 wypisz_droga()

```
void wypisz_droga (
            droga * pHead_droga )
```

Funkcja wypisuje liste drog

Parameters

pHead_droga	wskaznik na pierwszy element listy

Returns

Funkcja nie zwraca niczego.

Here is the caller graph for this function:

4.3.1.11 wypisz_miasta()

```
void wypisz_miasta (
            miasto * pHead,
            std::ostream & wyjscie )
```

Funkcja zapisuje trasy do pliku wyjsciowego

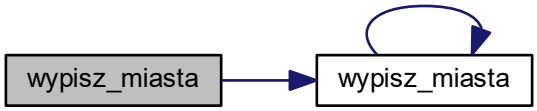
Parameters

pHead	wskaznik na pierwszy element listy
wyjscie	nazwa pliku wyjsciowego

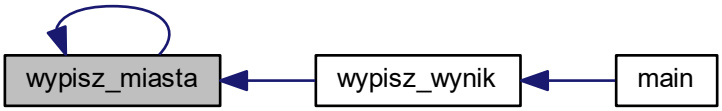
Returns

Funkcja nie zwraca niczego.

Here is the call graph for this function:



Here is the caller graph for this function:



4.3.1.12 wypisz_miasto()

```
void wypisz_miasto (
            miasto * pHead )
```

Funkcja wypisuje liste miast

Parameters

pHead	wskaznik na pierwszy element listy

Returns

Funkcja nie zwraca niczego.

Here is the call graph for this function:

4.3.1.13 wypisz_wynik()

```
void wypisz_wynik (
            miasto * pHead,
            const std::string & wyjscie )
```

Funkcja zapisuje wynik do pliku wyjsciowego

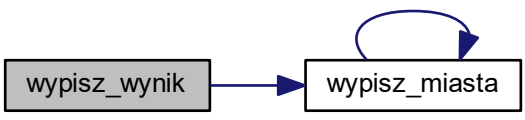
Parameters

in,out	*pHead*	wskaznik na pierwszy element listy
	wyjscie	nazwa pliku wyjsciowego

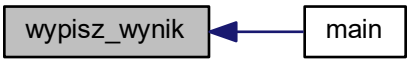
Returns

Funkcja nie zwraca niczego.

Here is the call graph for this function:



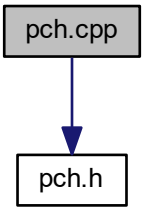
Here is the caller graph for this function:



4.4 pch.cpp File Reference

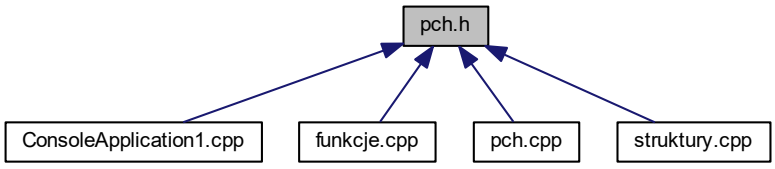
```
#include "pch.h"
```

Include dependency graph for pch.cpp:



4.5 pch.h File Reference

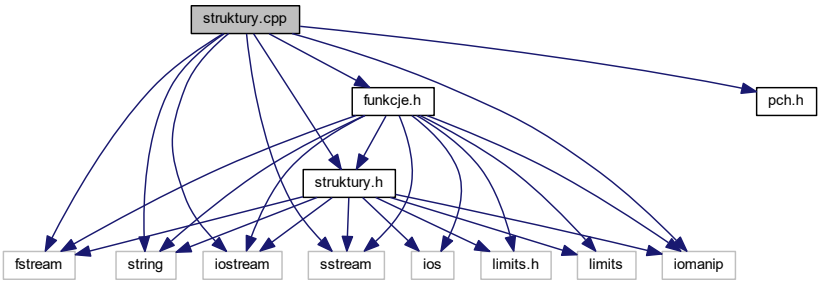
This graph shows which files directly or indirectly include this file:



4.6 struktury.cpp File Reference

```
#include <iostream>
#include <fstream>
#include <string>
#include <sstream>
#include <iomanip>
#include "pch.h"
#include "struktury.h"
#include "funkcje.h"
```

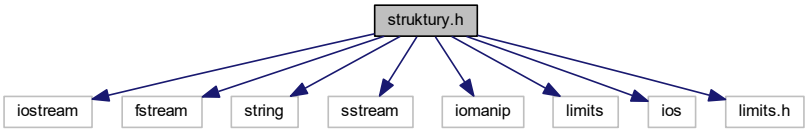
Include dependency graph for struktury.cpp:



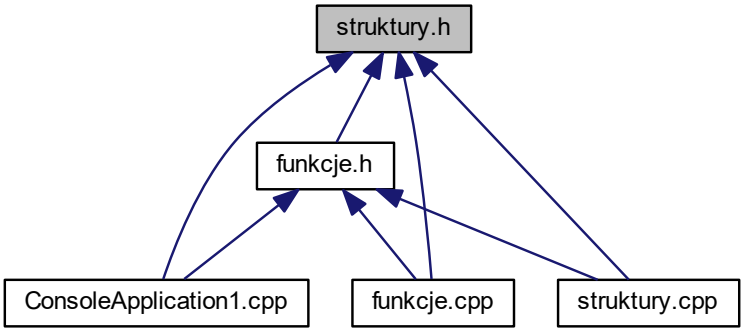
4.7 struktury.h File Reference

```
#include <iostream>
#include <fstream>
#include <string>
#include <sstream>
#include <iomanip>
#include <limits>
#include <ios>
#include <limits.h>
```

Include dependency graph for struktury.h:



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



Classes

- struct miasto
- struct droga