LAB2

0.1

Generated by Doxygen 1.8.6

Thu Mar 19 2015 01:29:59

Contents

Chapter 1

Class Index

1.1 Class List

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

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Chapter 3

Class Documentation

3.1 Kolejka< T > Class Template Reference

klasa **Kolejka** (p. 5) sluzy do wykonywania podstawowych operacji na Kolejce: dodaj,odejmij element. Przechowuje informacje o ilosci wszysktich elementow.

`#include <Kolejka.hh>`

Public Member Functions

- **Kolejka** ()
- ~**Kolejka** ()
- int **size** ()
- void **push_front** (T value)
- void **pop_back** ()
- void **show** ()

3.1.1 Detailed Description

template<typename T>class Kolejka< T >

klasa **Kolejka** (p. 5) sluzy do wykonywania podstawowych operacji na Kolejce: dodaj,odejmij element. Przechowuje informacje o ilosci wszysktich elementow.

Definition at line 21 of file Kolejka.hh.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 template<typename T > Kolejka< T >::Kolejka ( )

brief Konstruktor bezparametryczny

Konstruktor bezparametryczny, ustawia parametry na 0

Definition at line 59 of file Kolejka.hh.

3.1.2.2 template<typename T > Kolejka< T >::~Kolejka ( )

Destruktor, usuwa kolejne elementy kolejki zaczynajac od poczatku

Definition at line 69 of file Kolejka.hh.

3.1.3 Member Function Documentation

3.1.3.1 template<typename T > void Kolejka< T >::pop_back ( )

brief Funkcja zdejmuje element z konca kolejki

Funkcja usuwa element z konca kolejki

Precondition

Kolejka (p. 5) nie moze byc pusta

Definition at line 106 of file Kolejka.hh.

3.1.3.2 template<typename T > void Kolejka< T >::push_front (T *value*)

brief Funkcja dodaje element na poczatek kolejki

Funkcja sluzy do dodania elementu na poczatek kolejki

Parameters

in	*value-typ*	int, wartosc elementu zmiennej dodanej do kolejki

Definition at line 92 of file Kolejka.hh.

3.1.3.3 template<typename T > void Kolejka< T >::show ( )

brief Funkcja wyswietla wszystkie elementy na standardowe wyjscie

Funkcja wyswietla elementy kolejki

Definition at line 134 of file Kolejka.hh.

3.1.3.4 template<typename T > int Kolejka< T >::size ( )

brief Funkcja zwraca rozmiar kolejki

Returns

Funkcja zwraca wartosc rozmiaru kolejki

Definition at line 82 of file Kolejka.hh.

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

- inc/**Kolejka.hh**

3.2 List< T > Class Template Reference

klasa **List** (p. 6) sluzy do wykonywania podstawowych operacji na Liscie: dodaj,odejmij element. Przechowuje informacje o ilosci wszysktich elementow.

`#include <Lista.hh>`

Public Member Functions

- **List** ()

- ~**List** ()
- int **size** ()
- void **push_front** (T value)
- void **pop_front** ()
- void **push_back** (T value)
- void **pop_back** ()
- void **show** ()

3.2.1 Detailed Description

template<typename T>class List< T >

klasa **List** (p. 6) sluzy do wykonywania podstawowych operacji na Liscie: dodaj,odejmij element. Przechowuje informacje o ilosci wszysktich elementow.

Definition at line 23 of file Lista.hh.

3.2.2 Constructor & Destructor Documentation

3.2.2.1 template<typename T > List< T >::List ( )

brief Konstruktor bezparametryczny

Konstruktor bezparametryczny, ustawia parametry na 0

Definition at line 73 of file Lista.hh.

3.2.2.2 template<typename T > List< T >::~List ( )

brief Destruktor

Destruktor, usuwa kolejne elementy listy zaczynajac od poczatku

Definition at line 83 of file Lista.hh.

3.2.3 Member Function Documentation

3.2.3.1 template<typename T > void List< T >::pop_back ( )

brief Funkcja zdejmuje element z konca listy

Funkcja usuwa element z konca listy

Precondition

Lista nie moze byc pusta

Definition at line 159 of file Lista.hh.

3.2.3.2 template<typename T > void List< T >::pop_front ( )

brief Funkcja zdejmuje element z poczatku listy

Funkcja usuwa element z poczatku listy

Precondition

Lista nie moze byc pusta

Definition at line 120 of file Lista.hh.

3.2.3.3 template<typename T > void List< T >::push_back (T *value*)

brief Funkcja dodaje element na koniec listy

Funkcja dodaje element na koniec listy

Parameters

in	*value*	- typ int, wartosc elementu dodanego na koniec listy

Definition at line 138 of file Lista.hh.

3.2.3.4 template<typename T > void List< T >::push_front (T *value*)

brief Funkcja dodaje element na poczatek listy

Funkcja sluzy do dodania elementu na poczatek listy

Parameters

in	*value-typ*	int, wartosc elementu zmiennej dodanej do listy

Definition at line 106 of file Lista.hh.

3.2.3.5 template<typename T > void List< T >::show ( )

brief Funkcja wyswietla wszystkie elementy na standardowe wyjscie

Funkcja wyswietla elementy listy

Definition at line 187 of file Lista.hh.

3.2.3.6 template<typename T > int List< T >::size ( )

brief Funkcja zwraca rozmiar listy

Returns

Funkcja zwraca wartosc rozmiaru listy

Definition at line 96 of file Lista.hh.

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

- inc/**Lista.hh**

3.3 Node< T > Struct Template Reference

`#include <Kolejka.hh>`

Public Attributes

- T **val**
- **Node**< T > * **next**

3.3.1 Detailed Description

template<typename T>struct Node< T >

Definition at line 10 of file Kolejka.hh.

3.3.2 Member Data Documentation

3.3.2.1 template<typename T> Node<T>* Node< T >::next

Definition at line 13 of file Kolejka.hh.

3.3.2.2 template<typename T> T Node< T >::val

Definition at line 12 of file Kolejka.hh.

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

- inc/**Kolejka.hh**

3.4 NodeL< T > Struct Template Reference

`#include <Lista.hh>`

Public Attributes

- T **val**
- **NodeL**< T > * **next**

3.4.1 Detailed Description

template<typename T>struct NodeL< T >

Definition at line 11 of file Lista.hh.

3.4.2 Member Data Documentation

3.4.2.1 template<typename T> NodeL<T>* NodeL< T >::next

Definition at line 14 of file Lista.hh.

3.4.2.2 template<typename T> T NodeL< T >::val

Definition at line 13 of file Lista.hh.

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

- inc/**Lista.hh**

3.5 Stack< T > Class Template Reference

`#include <Stack.hh>`

Public Member Functions

- **Stack** (int **capacity**)
- void **push** (T value)
- T **peek** ()
- int **size** ()
- ~**Stack** ()
- void **pop** ()

Public Attributes

- T * **top**
- int **capacity**
- T * **storage**

3.5.1 Detailed Description

template<typename T>class Stack< T >

Definition at line 10 of file Stack.hh.

3.5.2 Constructor & Destructor Documentation

3.5.2.1 template<typename T > Stack< T >::Stack (int *capacity*)

Konstruktor parametryczny klasy **Stack** (p. 10)

Parameters

in	*capacity*	- typ int, rozmiar stosu

Definition at line 64 of file Stack.hh.

3.5.2.2 template<typename T > Stack< T >::~Stack ( )

Destruktor klasy **Stack** (p. 10)

Definition at line 122 of file Stack.hh.

3.5.3 Member Function Documentation

3.5.3.1 template<typename T > T Stack< T >::peek ( )

Funkcja pokazuje element znajdujacy sie na szczycie stosu

Precondition

Stos nie moze byc pusty

Definition at line 102 of file Stack.hh.

3.5.3.2 template<typename T > void Stack< T >::pop ( )

Funkcja pop zdejmuje ostatni element ze stosu

Precondition

Stos nie moze byc pusty

Definition at line 133 of file Stack.hh.

3.5.3.3 template<typename T > void Stack< T >::push (T *value*)

Funkcja dodaje element na koniec tablicy stosu

Parameters

in	*value*	- typ int, wartosc dodana do stosu

Postcondition

wykorzystana metoda podwajania do powiekszania stosu

Definition at line 81 of file Stack.hh.

3.5.3.4 template<typename T > int Stack< T >::size ( )

Returns

zwraca ilosc elementow stosu

Definition at line 113 of file Stack.hh.

3.5.4 Member Data Documentation

3.5.4.1 template<typename T> int Stack< T >::capacity

Definition at line 20 of file Stack.hh.

3.5.4.2 template<typename T> T* Stack< T >::storage

Definition at line 24 of file Stack.hh.

3.5.4.3 template<typename T> T* Stack< T >::top

Definition at line 16 of file Stack.hh.

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

- inc/**Stack.hh**

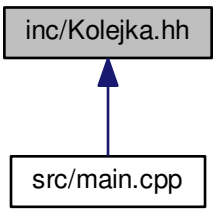
Chapter 4

File Documentation

4.1 inc/Kolejka.hh File Reference

Struktura przechowujaca wartosc wezla i wskaznik na nastepny element typu **Node** (p. 8).

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



Classes

- struct **Node**< **T** >
- class **Kolejka**< **T** >

  *klasa **Kolejka** (p. 5) sluzy do wykonywania podstawowych operacji na Kolejce: dodaj,odejmij element. Przechowuje informacje o ilosci wszysktich elementow.*

4.1.1 Detailed Description

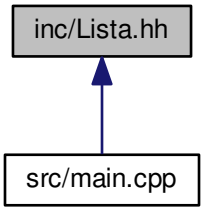
Struktura przechowujaca wartosc wezla i wskaznik na nastepny element typu **Node** (p. 8).

Definition in file **Kolejka.hh**.

4.2 inc/Lista.hh File Reference

Struktura przechowujaca wartosc wezla i wskaznik na nastepny element typu **Node** (p. 8).

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



Classes

- struct **NodeL**< **T** >
- class **List**< **T** >

  *klasa **List** (p. 6) sluzy do wykonywania podstawowych operacji na Liscie: dodaj,odejmij element. Przechowuje informacje o ilosci wszysktich elementow.*

4.2.1 Detailed Description

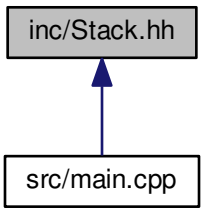
Struktura przechowujaca wartosc wezla i wskaznik na nastepny element typu **Node** (p. 8).

Definition in file **Lista.hh**.

4.3 inc/Stack.hh File Reference

Klasa **Stack** (p. 10) sluzy do przechowywania, dodawania,zdejmowania kolejnych elementow stosu.

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



Classes

- class **Stack**< **T** >

Functions

- template<typename T >
  std::ostream & **operator**<< (std::ostream &out, const **Stack**< T > &stack)

4.3.1 Detailed Description

Klasa **Stack** (p. 10) sluzy do przechowywania, dodawania,zdejmowania kolejnych elementow stosu.

Definition in file **Stack.hh**.

4.3.2 Function Documentation

4.3.2.1 template<typename T > std::ostream& operator<< (std::ostream & *out,* const Stack< T > & *stack*)

Funkcja operatorowa sluzy do wyswietlania stosu,zbedna

Parameters

in	*&out*	- referencja do strumienia wyjsciowego
in	*&stack-*	referencja do stosu

Returns

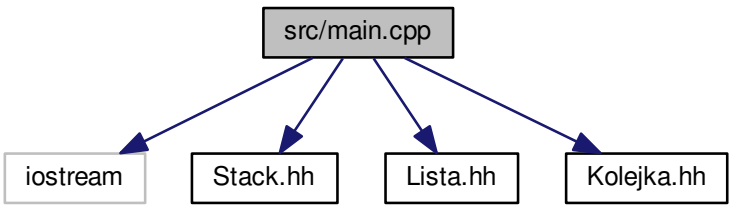
zwraca referencje do strumienia wyjsciowego

Definition at line 148 of file Stack.hh.

4.4 src/main.cpp File Reference

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

Include dependency graph for main.cpp:



Functions

- int **main** ()

4.4.1 Function Documentation

4.4.1.1 int main ( )

Definition at line 6 of file main.cpp.

Index