

## Tablica asocjacyjna

Wygenerowano przez Doxygen 1.8.4

So, 5 kwi 2014 19:01:28

## Spis treści

<b>1</b>	<b>Tablica asocjacyjne</b>	<b>1</b>
1.1	Opis programu . . . . .	1
1.2	Autor . . . . .	1
<b>2</b>	<b>Indeks klas</b>	<b>1</b>
2.1	Lista klas . . . . .	2
<b>3</b>	<b>Indeks plików</b>	<b>2</b>
3.1	Lista plików . . . . .	2
<b>4</b>	<b>Dokumentacja klas</b>	<b>2</b>
4.1	Dokumentacja szablonu klasy <code>Aarray&lt; Value &gt;</code> . . . . .	2
4.1.1	Dokumentacja funkcji składowych . . . . .	3
4.1.2	Dokumentacja atrybutów składowych . . . . .	3
4.2	Dokumentacja struktury <code>Aarray&lt; Value &gt;::Pair</code> . . . . .	4
4.2.1	Dokumentacja atrybutów składowych . . . . .	4
<b>5</b>	<b>Dokumentacja plików</b>	<b>4</b>
5.1	Dokumentacja pliku <code>aarray.hh</code> . . . . .	4
5.1.1	Opis szczegółowy . . . . .	4
5.2	Dokumentacja pliku <code>main.cpp</code> . . . . .	4
5.2.1	Opis szczegółowy . . . . .	5
5.2.2	Dokumentacja funkcji . . . . .	5
5.3	Dokumentacja pliku <code>mainpage.dox</code> . . . . .	5
	<b>Indeks</b>	<b>6</b>

## 1 Tablica asocjacyjne

### 1.1 Opis programu

Program umożliwiający operacje na tablicy asocjacyjnej - abstrakcyjnego typu danych, który przechowuje pary - unikatowy klucz oraz wartość. Operacje mogą być wykonywane w oparciu o wektor struktury danych `Pair`.

### 1.2 Autor

Program wykonała: Agnieszka Wisniewska, nr albumu: 200 466

## 2 Indeks klas

## 2.1 Lista klas

Tutaj znajdują się klasy, struktury, unie i interfejsy wraz z ich krótkimi opisami:

[Aarray< Value >](#) 2

[Aarray< Value >::Pair](#) 4

## 3 Indeks plików

### 3.1 Lista plików

Tutaj znajduje się lista wszystkich plików z ich krótkimi opisami:

[aarray.hh](#)  
Plik zawiera definicje szablonu klasy [Aarray](#) 4

[main.cpp](#)  
Plik zawierający główną funkcję programu 4

## 4 Dokumentacja klas

### 4.1 Dokumentacja szablonu klasy [Aarray< Value >](#)

```
#include <aarray.hh>
```

#### Komponenty

- struct [Pair](#)

#### Metody publiczne

- void [Add](#) (Value value, std::string key)  
*Funkcja dodająca parę klucz-wartość do tablicy.*
- void [Delete](#) (std::string key)  
*Funkcja usuwająca wybraną parę klucz-wartość z tablicy.*
- Value [GetValue](#) (std::string key)  
*Funkcja pobierająca wartość wybranego klucza.*
- void [ChangeValue](#) (Value newValue, std::string key)  
*Funkcja zmieniająca wartość pod wybranym kluczem.*
- unsigned int [Size](#) ()  
*Funkcja zwraca ilość par klucz-wartość*
- bool [IsEmpty](#) ()  
*Funkcja sprawdzająca, czy tablica jest pusta.*

#### Metody prywatne

- unsigned int [Search](#) (std::string key)

## Atrybuty prywatne

- std::vector< Pair > PairVec

## 4.1.1 Dokumentacja funkcji składowych

## 4.1.1.1 template&lt;class Value &gt; void Aarray&lt; Value &gt;::Add ( Value value, std::string key )

Funkcja dodająca parę klucz-wartość do tablicy.

## Parametry

<i>value</i>	wartość dla dodawanego klucza
<i>key</i>	unikatowy klucz, do którego przypisujemy wartość

## 4.1.1.2 template&lt;class Value &gt; void Aarray&lt; Value &gt;::ChangeValue ( Value newValue, std::string key )

Funkcja zmieniająca wartość pod wybranym kluczem.

## Parametry

<i>newValue</i>	nowa wartość dla klucza
<i>key</i>	unikatowy klucz, którego wartość zmieniamy

## 4.1.1.3 template&lt;class Value &gt; void Aarray&lt; Value &gt;::Delete ( std::string key )

Funkcja usuwająca wybraną parę klucz-wartość z tablicy.

## Parametry

<i>key</i>	klucz, który usuwamy wraz z jego wartością
------------	--

## 4.1.1.4 template&lt;class Value &gt; Value Aarray&lt; Value &gt;::GetValue ( std::string key )

Funkcja pobierająca wartość wybranego klucza.

## Parametry

<i>key</i>	klucz, którego wartość chcemy znać
------------	------------------------------------

## 4.1.1.5 template&lt;class Value &gt; bool Aarray&lt; Value &gt;::IsEmpty ( )

Funkcja sprawdzająca, czy tablica jest pusta.

## Zwraca

true jeśli tablica jest pusta  
false jeśli na tablicy są jakieś pary klucz-wartość

## 4.1.1.6 template&lt;class Value &gt; unsigned int Aarray&lt; Value &gt;::Search ( std::string key ) [private]

## 4.1.1.7 template&lt;class Value &gt; unsigned int Aarray&lt; Value &gt;::Size ( )

Funkcja zwraca ilość par klucz-wartość

## 4.1.2 Dokumentacja atrybutów składowych

## 4.1.2.1 template&lt;class Value&gt; std::vector&lt;Pair&gt; Aarray&lt; Value &gt;::PairVec [private]

Dokumentacja dla tej klasy została wygenerowana z pliku:

- [aarray.hh](#)

## 4.2 Dokumentacja struktury `Aarray< Value >::Pair`

Atrybuty publiczne

- `std::string` [key](#)
- `Value` [value](#)

### 4.2.1 Dokumentacja atrybutów składowych

#### 4.2.1.1 `template<class Value> std::string Aarray< Value >::Pair::key`

#### 4.2.1.2 `template<class Value> Value Aarray< Value >::Pair::value`

Dokumentacja dla tej struktury została wygenerowana z pliku:

- [aarray.hh](#)

## 5 Dokumentacja plików

### 5.1 Dokumentacja pliku `aarray.hh`

Plik zawiera definicje szablonu klasy [Aarray](#).

```
#include <vector>
#include <string>
```

Komponenty

- `class` [Aarray< Value >](#)
- `struct` [Aarray< Value >::Pair](#)

#### 5.1.1 Opis szczegółowy

Plik zawiera definicje szablonu klasy [Aarray](#).

### 5.2 Dokumentacja pliku `main.cpp`

Plik zawierający główną funkcję programu.

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

Funkcje

- `int` [main](#) ()

*Główna funkcja programu.*

**5.2.1 Opis szczegółowy**

Plik zawierający główną funkcję programu.

**5.2.2 Dokumentacja funkcji****5.2.2.1 int main ( )**

Główna funkcja programu.

Obecnie służy wyłącznie do testów.

**5.3 Dokumentacja pliku mainpage.dox**

## Skorowidz

### Aarray

- Add, [3](#)
- ChangeValue, [3](#)
- Delete, [3](#)
- GetValue, [3](#)
- IsEmpty, [3](#)
- PairVec, [3](#)
- Search, [3](#)
- Size, [3](#)

Aarray< Value >, [2](#)

Aarray< Value >::Pair, [4](#)

aarray.hh, [4](#)

Aarray::Pair

- key, [4](#)
- value, [4](#)

### Add

- Aarray, [3](#)

### ChangeValue

- Aarray, [3](#)

### Delete

- Aarray, [3](#)

### GetValue

- Aarray, [3](#)

### IsEmpty

- Aarray, [3](#)

### key

- Aarray::Pair, [4](#)

### main

- main.cpp, [5](#)

main.cpp, [4](#)

- main, [5](#)

mainpage.dox, [5](#)

### PairVec

- Aarray, [3](#)

### Search

- Aarray, [3](#)

### Size

- Aarray, [3](#)

### value

- Aarray::Pair, [4](#)