

My Project

Generated by Doxygen 1.10.0

| | |
|--|----------|
| 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 Studentas Class Reference | 7 |
| 4.1.1 Detailed Description | 8 |
| 4.1.2 Constructor & Destructor Documentation | 9 |
| 4.1.2.1 Studentas() [1/4] | 9 |
| 4.1.2.2 Studentas() [2/4] | 9 |
| 4.1.2.3 Studentas() [3/4] | 9 |
| 4.1.2.4 Studentas() [4/4] | 9 |
| 4.1.2.5 ~Studentas() | 9 |
| 4.1.3 Member Function Documentation | 9 |
| 4.1.3.1 addPazymys() | 9 |
| 4.1.3.2 getEgzaminoRezultatas() | 10 |
| 4.1.3.3 getGalutinisVid() | 10 |
| 4.1.3.4 getMediana() | 10 |
| 4.1.3.5 getPavarde() | 10 |
| 4.1.3.6 getPazymiai() | 10 |
| 4.1.3.7 getVardas() | 10 |
| 4.1.3.8 operator=() [1/2] | 11 |
| 4.1.3.9 operator=() [2/2] | 11 |
| 4.1.3.10 setEgzaminoRezultatas() | 11 |
| 4.1.3.11 setGalutinisVid() | 11 |
| 4.1.3.12 setMediana() | 11 |
| 4.1.3.13 setPavarde() | 12 |
| 4.1.3.14 setPazymiai() | 12 |
| 4.1.3.15 setVardas() | 12 |
| 4.1.4 Friends And Related Symbol Documentation | 12 |
| 4.1.4.1 operator<< | 12 |
| 4.1.4.2 operator>> | 13 |
| 4.2 Zmogus Class Reference | 13 |
| 4.2.1 Detailed Description | 14 |
| 4.2.2 Constructor & Destructor Documentation | 14 |
| 4.2.2.1 Zmogus() [1/2] | 14 |
| 4.2.2.2 Zmogus() [2/2] | 14 |
| 4.2.2.3 ~Zmogus() | 15 |

| | |
|--|-----------|
| 4.2.3 Member Function Documentation | 15 |
| 4.2.3.1 getPavarde() | 15 |
| 4.2.3.2 getVardas() | 15 |
| 4.2.3.3 setPavarde() | 15 |
| 4.2.3.4 setVardas() | 15 |
| 4.2.4 Member Data Documentation | 16 |
| 4.2.4.1 pavarde_ | 16 |
| 4.2.4.2 vardas_ | 16 |
| 5 File Documentation | 17 |
| 5.1 C:/Users/acer/Desktop/Adri/VSCode/Programavimas/Funkcijos.cpp File Reference | 17 |
| 5.1.1 Function Documentation | 18 |
| 5.1.1.1 ar_vargsiukas() | 18 |
| 5.1.1.2 failolsvedimasDeque1() | 18 |
| 5.1.1.3 failolsvedimasList1() | 18 |
| 5.1.1.4 failolsvedimasVector1() | 18 |
| 5.1.1.5 generuotiFaila() | 18 |
| 5.1.1.6 mediana() | 18 |
| 5.1.1.7 nuskaitytiFailaDeque1() | 19 |
| 5.1.1.8 nuskaitytiFailaDeque2() | 19 |
| 5.1.1.9 nuskaitytiFailaDeque3() | 19 |
| 5.1.1.10 nuskaitytiFailaList1() | 19 |
| 5.1.1.11 nuskaitytiFailaList2() | 19 |
| 5.1.1.12 nuskaitytiFailaList3() | 19 |
| 5.1.1.13 nuskaitytiFailaVector1() | 19 |
| 5.1.1.14 nuskaitytiFailaVector2() | 20 |
| 5.1.1.15 nuskaitytiFailaVector3() | 20 |
| 5.1.1.16 palyginti_pagal_galutini_vidurki() | 20 |
| 5.1.1.17 palyginti_pagal_galutini_vidurki_didejimo_tvarka() | 20 |
| 5.1.1.18 palyginti_pagal_mediana() | 20 |
| 5.1.1.19 palyginti_pagal_pavarde() | 20 |
| 5.1.1.20 palyginti_pagal_varda() | 20 |
| 5.1.1.21 RezultatuVaizdavimas() | 20 |
| 5.1.1.22 testai() | 21 |
| 5.1.1.23 vidurkis_galutinis() | 21 |
| 5.2 C:/Users/acer/Desktop/Adri/VSCode/Programavimas/funkcijos.h File Reference | 21 |
| 5.2.1 Function Documentation | 22 |
| 5.2.1.1 ar_vargsiukas() | 22 |
| 5.2.1.2 failolsvedimasDeque1() | 22 |
| 5.2.1.3 failolsvedimasList1() | 22 |
| 5.2.1.4 failolsvedimasVector1() | 22 |
| 5.2.1.5 generuotiFaila() | 22 |

| | |
|--|-----------|
| 5.2.1.6 mediana() | 22 |
| 5.2.1.7 nuskaitytiFailaDeque1() | 22 |
| 5.2.1.8 nuskaitytiFailaDeque2() | 23 |
| 5.2.1.9 nuskaitytiFailaDeque3() | 23 |
| 5.2.1.10 nuskaitytiFailaList1() | 23 |
| 5.2.1.11 nuskaitytiFailaList2() | 23 |
| 5.2.1.12 nuskaitytiFailaList3() | 23 |
| 5.2.1.13 nuskaitytiFailaVector1() | 23 |
| 5.2.1.14 nuskaitytiFailaVector2() | 23 |
| 5.2.1.15 nuskaitytiFailaVector3() | 24 |
| 5.2.1.16 palyginti_pagal_galutini_vidurki() | 24 |
| 5.2.1.17 palyginti_pagal_galutini_vidurki_didejimo_tvarka() | 24 |
| 5.2.1.18 palyginti_pagal_mediana() | 24 |
| 5.2.1.19 palyginti_pagal_pavarde() | 24 |
| 5.2.1.20 palyginti_pagal_varda() | 24 |
| 5.2.1.21 RezultatuVaizdavimas() | 24 |
| 5.2.1.22 testai() | 24 |
| 5.2.1.23 vidurkis_galutinis() | 25 |
| 5.3 funkcijos.h | 25 |
| 5.4 C:/Users/acer/Desktop/Adri/VSCode/Programavimas/studentas.h File Reference | 25 |
| 5.5 studentas.h | 26 |
| 5.6 C:/Users/acer/Desktop/Adri/VSCode/Programavimas/Vektoriai.cpp File Reference | 28 |
| 5.6.1 Function Documentation | 28 |
| 5.6.1.1 main() | 28 |
| 5.6.2 Variable Documentation | 29 |
| 5.6.2.1 m | 29 |
| 5.6.2.2 MAX_ND | 29 |
| 5.6.2.3 MAX_STUDENTU | 29 |
| 5.6.2.4 menu | 29 |
| 5.6.2.5 n | 29 |
| 5.6.2.6 pasirinkimas | 29 |
| 5.6.2.7 raide | 29 |
| 5.6.2.8 suma | 29 |
| 5.6.2.9 variantas_namu_darbas | 29 |
| 5.6.2.10 variantas_studentas | 29 |
| 5.6.2.11 vidurkis | 29 |
| Index | 31 |

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

| | |
|---------------------|--------------------|
| Zmogus | 13 |
| Studentas | 7 |

Chapter 2

Class Index

2.1 Class List

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

| | | |
|---------------------------|--|--------------------|
| Studentas | Studento klasė, paveldinti žmogaus savybes | 7 |
| Zmogus | Bazinė klasė, aprašanti žmogų | 13 |

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

| | |
|--|----|
| C:/Users/acer/Desktop/Adri/VSCode/Programavimas/ Funkcijos.cpp | 17 |
| C:/Users/acer/Desktop/Adri/VSCode/Programavimas/ funkcijos.h | 21 |
| C:/Users/acer/Desktop/Adri/VSCode/Programavimas/ studentas.h | 25 |
| C:/Users/acer/Desktop/Adri/VSCode/Programavimas/ Vektoriai.cpp | 28 |

Chapter 4

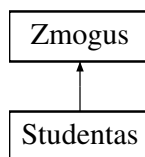
Class Documentation

4.1 Studentas Class Reference

Studento klasė, paveldinti žmogaus savybes.

```
#include <studentas.h>
```

Inheritance diagram for Studentas:



Public Member Functions

- [Studentas](#) ()
Default konstruktorius.
- [Studentas](#) (const vector< int > &pazymiai, const string &vardas, const string &pavarde, int egzamino_ ← rezultatas, double galutinis_vid, double [mediana](#))
- [Studentas](#) (const [Studentas](#) &other)
Copy konstruktorius.
- [Studentas](#) & [operator=](#) (const [Studentas](#) &other)
Copy assignment operatorius.
- [Studentas](#) ([Studentas](#) &&other) noexcept
Move konstruktorius.
- [Studentas](#) & [operator=](#) ([Studentas](#) &&other) noexcept
Move assignment operatorius.
- [~Studentas](#) ()
Destruktorius.
- void [setVardas](#) (const string &vardas) override
Nustato studento vardą.
- void [setPavarde](#) (const string &pavarde) override
Nustato studento pavardę.
- void [setPazymiai](#) (const vector< int > &pazymiai)

- Nustato studento pažymius.*
 - void [setEgzaminoRezultatas](#) (int egzamino_rezultatas)
- Nustato studento egzamino rezultata.*
 - void [setGalutinisVid](#) (double galutinis_vid)
- Nustato studento galutinį vidurkį.*
 - void [setMediana](#) (double [mediana](#))
- Nustato studento medianą.*
 - string [getVardas](#) () const override
- Grąžina studento vardą.*
 - string [getPavarde](#) () const override
- Grąžina studento pavardę.*
 - const vector< int > & [getPazymiai](#) () const
- Grąžina studento pažymius.*
 - int [getEgzaminoRezultatas](#) () const
- Grąžina studento egzamino rezultata.*
 - double [getGalutinisVid](#) () const
- Grąžina studento galutinį vidurkį.*
 - double [getMediana](#) () const
- Grąžina studento medianą.*
 - void [addPazymys](#) (int pazymys)
- Prideda naują pažymį studentui.*

Public Member Functions inherited from [Zmogus](#)

- [Zmogus](#) ()=default
- Default konstruktorius.*
- [Zmogus](#) (const string &vardas, const string &pavarde)
- Klasės konstruktorius su parametrais.*
- virtual [~Zmogus](#) ()
- Default destruktoriaus.*

Friends

- istream & [operator>>](#) (istream &is, [Studentas](#) &studentas)
- Skaito studento duomenis.*
- ostream & [operator<<](#) (ostream &os, const [Studentas](#) &studentas)
- Spausdina studento duomenis.*

Additional Inherited Members

Protected Attributes inherited from [Zmogus](#)

- string [vardas_](#)
- string [pavarde_](#)

4.1.1 Detailed Description

Studento klasė, paveldinti žmogaus savybes.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 Studentas() [1/4]

```
Studentas::Studentas ( ) [inline]
```

Default konstruktorius.

4.1.2.2 Studentas() [2/4]

```
Studentas::Studentas (
    const vector< int > & pazymiai,
    const string & vardas,
    const string & pavarde,
    int egzamino_rezultatas,
    double galutinis_vid,
    double mediana ) [inline]
```

4.1.2.3 Studentas() [3/4]

```
Studentas::Studentas (
    const Studentas & other ) [inline]
```

Copy konstruktorius.

4.1.2.4 Studentas() [4/4]

```
Studentas::Studentas (
    Studentas && other ) [inline], [noexcept]
```

Move konstruktorius.

4.1.2.5 ~Studentas()

```
Studentas::~~Studentas ( ) [inline]
```

Destruktorius.

4.1.3 Member Function Documentation

4.1.3.1 addPazymys()

```
void Studentas::addPazymys (
    int pazymys ) [inline]
```

Prideda naują pažymį studentui.

Parameters

| | |
|----------------|----------------|
| <i>pazymys</i> | Naujas pažymys |
|----------------|----------------|

4.1.3.2 getEgzaminoRezultatas()

```
int Studentas::getEgzaminoRezultatas ( ) const [inline]
```

Grąžina studento egzamino rezultatą.

4.1.3.3 getGalutinisVid()

```
double Studentas::getGalutinisVid ( ) const [inline]
```

Grąžina studento galutinį vidurkį.

4.1.3.4 getMediana()

```
double Studentas::getMediana ( ) const [inline]
```

Grąžina studento medianą.

4.1.3.5 getPavarde()

```
string Studentas::getPavarde ( ) const [inline], [override], [virtual]
```

Grąžina studento pavardę.

Implements [Zmogus](#).

4.1.3.6 getPazymiai()

```
const vector< int > & Studentas::getPazymiai ( ) const [inline]
```

Grąžina studento pažymius.

4.1.3.7 getVardas()

```
string Studentas::getVardas ( ) const [inline], [override], [virtual]
```

Grąžina studento vardą.

Implements [Zmogus](#).

4.1.3.8 operator=() [1/2]

```
Studentas & Studentas::operator= (
    const Studentas & other ) [inline]
```

Copy assignment operatorius.

4.1.3.9 operator=() [2/2]

```
Studentas & Studentas::operator= (
    Studentas && other ) [inline], [noexcept]
```

Move assignment operatorius.

4.1.3.10 setEgzaminoRezultatas()

```
void Studentas::setEgzaminoRezultatas (
    int egzamino_rezultatas ) [inline]
```

Nustato studento egzamino rezultatą.

Parameters

| | |
|----------------------------|------------------------------|
| <i>egzamino_rezultatas</i> | Studento egzamino rezultatas |
|----------------------------|------------------------------|

4.1.3.11 setGalutinisVid()

```
void Studentas::setGalutinisVid (
    double galutinis_vid ) [inline]
```

Nustato studento galutinį vidurkį.

Parameters

| | |
|----------------------|-----------------------------|
| <i>galutinis_vid</i> | Studento galutinis vidurkis |
|----------------------|-----------------------------|

4.1.3.12 setMediana()

```
void Studentas::setMediana (
    double mediana ) [inline]
```

Nustato studento medianą.

Parameters

| | |
|----------------|------------------|
| <i>mediana</i> | Studento mediana |
|----------------|------------------|

4.1.3.13 setPavarde()

```
void Studentas::setPavarde (
    const string & pavarde ) [inline], [override], [virtual]
```

Nustato studento pavardę.

Parameters

| | |
|----------------|------------------|
| <i>pavarde</i> | Studento pavardė |
|----------------|------------------|

Reimplemented from [Zmogus](#).

4.1.3.14 setPazymiai()

```
void Studentas::setPazymiai (
    const vector< int > & pazymiai ) [inline]
```

Nustato studento pažymius.

Parameters

| | |
|-----------------|-------------------|
| <i>pazymiai</i> | Studento pažymiai |
|-----------------|-------------------|

4.1.3.15 setVardas()

```
void Studentas::setVardas (
    const string & vardas ) [inline], [override], [virtual]
```

Nustato studento vardą.

Parameters

| | |
|---------------|-----------------|
| <i>vardas</i> | Studento vardas |
|---------------|-----------------|

Reimplemented from [Zmogus](#).

4.1.4 Friends And Related Symbol Documentation

4.1.4.1 operator<<

```
ostream & operator<< (
    ostream & os,
    const Studentas & studentas ) [friend]
```

Spausdina studento duomenis.

Parameters

| | |
|------------------|--|
| <i>os</i> | Išvesties srautas |
| <i>studentas</i> | Studento objektas, kurio duomenys bus išspausdinti |

Returns

Išvesties srautas

4.1.4.2 operator>>

```
istream & operator>> (
    istream & is,
    Studentas & studentas ) [friend]
```

Skaito studento duomenis.

Parameters

| | |
|------------------|--|
| <i>is</i> | Išvesties srautas |
| <i>studentas</i> | Studento objektas, į kurį bus įrašyti duomenys |

Returns

Išvesties srautas

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

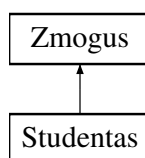
- C:/Users/acer/Desktop/Adri/VSCode/Programavimas/[studentas.h](#)

4.2 Zmogus Class Reference

Bazinė klasė, aprašanti žmogų.

```
#include <studentas.h>
```

Inheritance diagram for Zmogus:



Public Member Functions

- [Zmogus](#) ()=default
Default konstruktorius.
- [Zmogus](#) (const string &vardas, const string &pavarde)
Klasės konstruktorius su parametrais.
- virtual [~Zmogus](#) ()
Default destruktorius.
- virtual void [setVardas](#) (const string &vardas)
Nustato žmogaus vardą.
- virtual void [setPavarde](#) (const string &pavarde)
Nustato žmogaus pavardę.
- virtual string [getVardas](#) () const =0
Grąžina žmogaus vardą.
- virtual string [getPavarde](#) () const =0
Grąžina žmogaus pavardę.

Protected Attributes

- string [vardas_](#)
- string [pavarde_](#)

4.2.1 Detailed Description

Bazinė klasė, aprašanti žmogų.

4.2.2 Constructor & Destructor Documentation

4.2.2.1 Zmogus() [1/2]

```
Zmogus::Zmogus ( ) [default]
```

Default konstruktorius.

4.2.2.2 Zmogus() [2/2]

```
Zmogus::Zmogus (
    const string & vardas,
    const string & pavarde ) [inline]
```

Klasės konstruktorius su parametrais.

Parameters

| | |
|----------------|-----------------|
| <i>vardas</i> | Žmogaus vardas |
| <i>pavarde</i> | Žmogaus pavardė |

4.2.2.3 ~Zmogus()

```
virtual Zmogus::~~Zmogus ( ) [inline], [virtual]
```

Default destruktoriaus.

4.2.3 Member Function Documentation

4.2.3.1 getPavarde()

```
virtual string Zmogus::getPavarde ( ) const [pure virtual]
```

Grąžina žmogaus pavardę.

Implemented in [Studentas](#).

4.2.3.2 getVardas()

```
virtual string Zmogus::getVardas ( ) const [pure virtual]
```

Grąžina žmogaus vardą.

Implemented in [Studentas](#).

4.2.3.3 setPavarde()

```
virtual void Zmogus::setPavarde (
    const string & pavarde ) [inline], [virtual]
```

Nustato žmogaus pavardę.

Parameters

| | |
|----------------|-----------------|
| <i>pavarde</i> | Žmogaus pavardė |
|----------------|-----------------|

Reimplemented in [Studentas](#).

4.2.3.4 setVardas()

```
virtual void Zmogus::setVardas (
    const string & vardas ) [inline], [virtual]
```

Nustato žmogaus vardą.

Parameters

| | |
|---------------|----------------|
| <i>vardas</i> | Žmogaus vardas |
|---------------|----------------|

Reimplemented in [Studentas](#).

4.2.4 Member Data Documentation

4.2.4.1 pavarde_

```
string Zmogus::pavarde_ [protected]
```

Pavardės kintamasis

4.2.4.2 vardas_

```
string Zmogus::vardas_ [protected]
```

Vardo kintamasis

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

- C:/Users/acer/Desktop/Adri/VSCode/Programavimas/[studentas.h](#)

Chapter 5

File Documentation

5.1 C:/Users/acer/Desktop/Adri/VSCode/Programavimas/Funkcijos.cpp File Reference

```
#include "funkcijos.h"
#include <iostream>
#include <iomanip>
#include <algorithm>
#include <sstream>
#include <fstream>
#include <chrono>
#include <list>
#include <deque>
#include <cassert>
```

Functions

- double [mediana](#) (vector< int > pazymiai, int egzamino_rezultatas)
- double [vidurkis_galutinis](#) (double suma, int n, int egzamino_rezultatas)
- bool [palyginti_pagal_varda](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- bool [palyginti_pagal_pavarde](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- bool [palyginti_pagal_mediana](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- bool [palyginti_pagal_galutini_vidurki](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- bool [palyginti_pagal_galutini_vidurki_didejimo_tvarka](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- bool [ar_vargsiukas](#) (const [Studentas](#) &student)
- void [RezultatuVaizdavimas](#) (const vector< [Studentas](#) > &studentai, int pasirinkimas1)
- void [generuotiFaila](#) (string failoPavadinimas, int ndSkaicius, int studentuSkaicius)
- void [failoIsvedimasVector1](#) (const vector< [Studentas](#) > &studentai, string failoPavadinimas)
- void [nuskaitytiFailaVector1](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas, string kietakiuFailoPavadinimas)
- void [failoIsvedimasList1](#) (const list< [Studentas](#) > &studentai, string failoPavadinimas)
- void [nuskaitytiFailaList1](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas, string kietakiuFailoPavadinimas)
- void [failoIsvedimasDeque1](#) (const deque< [Studentas](#) > &studentai, string failoPavadinimas)
- void [nuskaitytiFailaDeque1](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas, string kietakiuFailoPavadinimas)

- void [nuskaitytiFailaVector2](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas)
- void [nuskaitytiFailaList2](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas)
- void [nuskaitytiFailaDeque2](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas)
- void [nuskaitytiFailaVector3](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas)
- void [nuskaitytiFailaList3](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas)
- void [nuskaitytiFailaDeque3](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas)
- void [testai](#) ()

5.1.1 Function Documentation

5.1.1.1 ar_vargsiukas()

```
bool ar_vargsiukas (  
    const Studentas & student )
```

5.1.1.2 failoIsvedimasDeque1()

```
void failoIsvedimasDeque1 (  
    const deque< Studentas > & studentai,  
    string failoPavadinimas )
```

5.1.1.3 failoIsvedimasList1()

```
void failoIsvedimasList1 (  
    const list< Studentas > & studentai,  
    string failoPavadinimas )
```

5.1.1.4 failoIsvedimasVector1()

```
void failoIsvedimasVector1 (  
    const vector< Studentas > & studentai,  
    string failoPavadinimas )
```

5.1.1.5 generuotiFaila()

```
void generuotiFaila (  
    string failoPavadinimas,  
    int ndSkaicius,  
    int studentuSkaicius )
```

5.1.1.6 mediana()

```
double mediana (  
    vector< int > pazymiai,  
    int egzamino_rezultatas )
```


5.1.1.7 nuskaitytiFailaDeque1()

```
void nuskaitytiFailaDeque1 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas,
    string kietakiuFailoPavadinimas )
```

5.1.1.8 nuskaitytiFailaDeque2()

```
void nuskaitytiFailaDeque2 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas )
```

5.1.1.9 nuskaitytiFailaDeque3()

```
void nuskaitytiFailaDeque3 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas )
```

5.1.1.10 nuskaitytiFailaList1()

```
void nuskaitytiFailaList1 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas,
    string kietakiuFailoPavadinimas )
```

5.1.1.11 nuskaitytiFailaList2()

```
void nuskaitytiFailaList2 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas )
```

5.1.1.12 nuskaitytiFailaList3()

```
void nuskaitytiFailaList3 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas )
```

5.1.1.13 nuskaitytiFailaVector1()

```
void nuskaitytiFailaVector1 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas,
    string kietakiuFailoPavadinimas )
```

5.1.1.14 nuskaitytiFailaVector2()

```
void nuskaitytiFailaVector2 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas )
```

5.1.1.15 nuskaitytiFailaVector3()

```
void nuskaitytiFailaVector3 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas )
```

5.1.1.16 palyginti_pagal_galutini_vidurki()

```
bool palyginti_pagal_galutini_vidurki (
    const Studentas & a,
    const Studentas & b )
```

5.1.1.17 palyginti_pagal_galutini_vidurki_didejimo_tvarka()

```
bool palyginti_pagal_galutini_vidurki_didejimo_tvarka (
    const Studentas & a,
    const Studentas & b )
```

5.1.1.18 palyginti_pagal_mediana()

```
bool palyginti_pagal_mediana (
    const Studentas & a,
    const Studentas & b )
```

5.1.1.19 palyginti_pagal_pavarde()

```
bool palyginti_pagal_pavarde (
    const Studentas & a,
    const Studentas & b )
```

5.1.1.20 palyginti_pagal_varda()

```
bool palyginti_pagal_varda (
    const Studentas & a,
    const Studentas & b )
```

5.1.1.21 RezultatuVaizdavimas()

```
void RezultatuVaizdavimas (
    const vector< Studentas > & studentai,
    int pasirinkimas1 )
```

5.1.1.22 testai()

```
void testai ( )
```

5.1.1.23 vidurkis_galutinis()

```
double vidurkis_galutinis (
    double suma,
    int n,
    int egzamino_rezultatas )
```

5.2 C:/Users/acer/Desktop/Adri/VSCode/Programavimas/funkcijos.h File Reference

```
#include <vector>
#include <list>
#include <deque>
#include "studentas.h"
```

Functions

- double [mediana](#) (vector< int > pazymiai, int egzamino_rezultatas)
- double [vidurkis_galutinis](#) (double [suma](#), int [n](#), int egzamino_rezultatas)
- bool [palyginti_pagal_varda](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- bool [palyginti_pagal_pavarde](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- bool [palyginti_pagal_mediana](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- bool [palyginti_pagal_galutini_vidurki](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- bool [palyginti_pagal_galutini_vidurki_didejimo_tvarka](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- bool [ar_vargsiukas](#) (const [Studentas](#) &student)
- void [RezultatuVaizdavimas](#) (const vector< [Studentas](#) > &studentai, int pasirinkimas1)
- void [generuotiFaila](#) (string failoPavadinimas, int ndSkaicius, int studentuSkaicius)
- void [failolsvedimasVector1](#) (const vector< [Studentas](#) > &studentai, string failoPavadinimas)
- void [nuskaitytiFailaVector1](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas, string kietakiauFailoPavadinimas)
- void [failolsvedimasList1](#) (const list< [Studentas](#) > &studentai, string failoPavadinimas)
- void [nuskaitytiFailaList1](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas, string kietakiauFailoPavadinimas)
- void [failolsvedimasDeque1](#) (const deque< [Studentas](#) > &studentai, string failoPavadinimas)
- void [nuskaitytiFailaDeque1](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas, string kietakiauFailoPavadinimas)
- void [nuskaitytiFailaVector2](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas)
- void [nuskaitytiFailaList2](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas)
- void [nuskaitytiFailaDeque2](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas)
- void [nuskaitytiFailaVector3](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas)
- void [nuskaitytiFailaList3](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas)
- void [nuskaitytiFailaDeque3](#) (string failoPavadinimas, string vargsiukuFailoPavadinimas)
- void [testai](#) ()

5.2.1 Function Documentation

5.2.1.1 ar_vargsiukas()

```
bool ar_vargsiukas (
    const Studentas & student )
```

5.2.1.2 failoIsvedimasDeque1()

```
void failoIsvedimasDeque1 (
    const deque< Studentas > & studentai,
    string failoPavadinimas )
```

5.2.1.3 failoIsvedimasList1()

```
void failoIsvedimasList1 (
    const list< Studentas > & studentai,
    string failoPavadinimas )
```

5.2.1.4 failoIsvedimasVector1()

```
void failoIsvedimasVector1 (
    const vector< Studentas > & studentai,
    string failoPavadinimas )
```

5.2.1.5 generuotiFaila()

```
void generuotiFaila (
    string failoPavadinimas,
    int ndSkaicius,
    int studentuSkaicius )
```

5.2.1.6 mediana()

```
double mediana (
    vector< int > pazymiai,
    int egzamino_rezultatas )
```

5.2.1.7 nuskaitytiFailaDeque1()

```
void nuskaitytiFailaDeque1 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas,
    string kietakiuFailoPavadinimas )
```

5.2.1.8 nuskaitytiFailaDeque2()

```
void nuskaitytiFailaDeque2 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas )
```

5.2.1.9 nuskaitytiFailaDeque3()

```
void nuskaitytiFailaDeque3 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas )
```

5.2.1.10 nuskaitytiFailaList1()

```
void nuskaitytiFailaList1 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas,
    string kietakiuFailoPavadinimas )
```

5.2.1.11 nuskaitytiFailaList2()

```
void nuskaitytiFailaList2 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas )
```

5.2.1.12 nuskaitytiFailaList3()

```
void nuskaitytiFailaList3 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas )
```

5.2.1.13 nuskaitytiFailaVector1()

```
void nuskaitytiFailaVector1 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas,
    string kietakiuFailoPavadinimas )
```

5.2.1.14 nuskaitytiFailaVector2()

```
void nuskaitytiFailaVector2 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas )
```

5.2.1.15 nuskaitytiFailaVector3()

```
void nuskaitytiFailaVector3 (
    string failoPavadinimas,
    string vargsiukuFailoPavadinimas )
```

5.2.1.16 palyginti_pagal_galutini_vidurki()

```
bool palyginti_pagal_galutini_vidurki (
    const Studentas & a,
    const Studentas & b )
```

5.2.1.17 palyginti_pagal_galutini_vidurki_didejimo_tvarka()

```
bool palyginti_pagal_galutini_vidurki_didejimo_tvarka (
    const Studentas & a,
    const Studentas & b )
```

5.2.1.18 palyginti_pagal_mediana()

```
bool palyginti_pagal_mediana (
    const Studentas & a,
    const Studentas & b )
```

5.2.1.19 palyginti_pagal_pavarde()

```
bool palyginti_pagal_pavarde (
    const Studentas & a,
    const Studentas & b )
```

5.2.1.20 palyginti_pagal_varda()

```
bool palyginti_pagal_varda (
    const Studentas & a,
    const Studentas & b )
```

5.2.1.21 RezultatuVaizdavimas()

```
void RezultatuVaizdavimas (
    const vector< Studentas > & studentai,
    int pasirinkimas1 )
```

5.2.1.22 testai()

```
void testai ( )
```

5.2.1.23 vidurkis_galutinis()

```
double vidurkis_galutinis (
    double suma,
    int n,
    int egzamino_rezultatas )
```

5.3 funkcijos.h

[Go to the documentation of this file.](#)

```
00001 #ifndef FUNKCIJOS_H
00002 #define FUNKCIJOS_H
00003
00004 #include <vector>
00005 #include <list>
00006 #include <deque>
00007 #include "studentas.h"
00008
00009 using namespace std;
00010
00011 double mediana(vector<int> pazymiai, int egzamino_rezultatas);
00012 double vidurkis_galutinis(double suma, int n, int egzamino_rezultatas);
00013 bool palyginti_pagal_varda(const Studentas &a, const Studentas &b);
00014 bool palyginti_pagal_pavarde(const Studentas &a, const Studentas &b);
00015 bool palyginti_pagal_mediana(const Studentas &a, const Studentas &b);
00016 bool palyginti_pagal_galutini_vidurki(const Studentas &a, const Studentas &b);
00017 bool palyginti_pagal_galutini_vidurki_didejimo_tvarka(const Studentas &a, const Studentas &b);
00018 bool ar_vargsiukas(const Studentas& student);
00019 void RezultatuVaizdavimas (const vector<Studentas>& studentai, int pasirinkimas1);
00020 void generuotiFaila(string failoPavadinimas, int ndSkaicius, int studentuSkaicius);
00021
00022 // 1 strategija
00023 void failoIsvedimasVector1(const vector<Studentas>& studentai, string failoPavadinimas);
00024 void nuskaitytiFailaVector1(string failoPavadinimas, string vargsiukuFailoPavadinimas, string
kietakiuFailoPavadinimas);
00025 void failoIsvedimasList1(const list<Studentas>& studentai, string failoPavadinimas);
00026 void nuskaitytiFailaList1(string failoPavadinimas, string vargsiukuFailoPavadinimas, string
kietakiuFailoPavadinimas);
00027 void failoIsvedimasDeque1(const deque<Studentas>& studentai, string failoPavadinimas);
00028 void nuskaitytiFailaDeque1(string failoPavadinimas, string vargsiukuFailoPavadinimas, string
kietakiuFailoPavadinimas);
00029
00030 // 2 strategija
00031 void nuskaitytiFailaVector2(string failoPavadinimas, string vargsiukuFailoPavadinimas);
00032 void nuskaitytiFailaList2(string failoPavadinimas, string vargsiukuFailoPavadinimas);
00033 void nuskaitytiFailaDeque2(string failoPavadinimas, string vargsiukuFailoPavadinimas);
00034
00035 // 3 strategija
00036 void nuskaitytiFailaVector3(string failoPavadinimas, string vargsiukuFailoPavadinimas);
00037 void nuskaitytiFailaList3(string failoPavadinimas, string vargsiukuFailoPavadinimas);
00038 void nuskaitytiFailaDeque3(string failoPavadinimas, string vargsiukuFailoPavadinimas);
00039
00040 void testai();
00041
00042 #endif
```

5.4 C:/Users/acer/Desktop/Adri/VSCoDe/Programavimas/studentas.h File Reference

```
#include <vector>
#include <string>
#include <limits>
#include <iostream>
```

Classes

- class [Zmogus](#)
Bazinė klasė, aprašanti žmogų.
- class [Studentas](#)
Studento klasė, paveldinti žmogaus savybes.

5.5 studentas.h

[Go to the documentation of this file.](#)

```
00001 #ifndef STUDENTAS_H
00002 #define STUDENTAS_H
00003
00004 #include <vector>
00005 #include <string>
00006 #include <limits>
00007 #include <iostream>
00008
00009 using namespace std;
00010
00015 class Zmogus {
00016 protected:
00017     string vardas_;
00018     string pavarde_;
00020 public:
00021
00025     Zmogus() = default;
00026
00032     Zmogus(const string& vardas, const string& pavarde) : vardas_(vardas), pavarde_(pavarde) {}
00033
00037     virtual ~Zmogus() {}
00038
00043     virtual void setVardas(const string& vardas) { vardas_ = vardas; }
00044
00049     virtual void setPavarde(const string& pavarde) { pavarde_ = pavarde; }
00050
00054     virtual string getVardas() const = 0;
00055
00059     virtual string getPavarde() const = 0;
00060 };
00061
00066 class Studentas : public Zmogus {
00067 private:
00068     vector<int> pazymiai_;
00069     int egzamino_rezultatas_;
00070     double galutinis_vid_;
00071     double mediana_;
00073 public:
00074
00079     Studentas() : egzamino_rezultatas_(0), galutinis_vid_(0), mediana_(0) {}
00080     Studentas(const vector<int>& pazymiai, const string& vardas, const string& pavarde, int
egzamino_rezultatas, double galutinis_vid, double mediana)
00081         : Zmogus(vardas, pavarde), pazymiai_(pazymiai), egzamino_rezultatas_(egzamino_rezultatas),
galutinis_vid_(galutinis_vid), mediana_(mediana) {}
00082
00087     // Copy constructor
00088     Studentas(const Studentas& other)
00089         : Zmogus(other.getVardas(), other.getPavarde()), pazymiai_(other.pazymiai_),
egzamino_rezultatas_(other.egzamino_rezultatas_),
00090         galutinis_vid_(other.galutinis_vid_), mediana_(other.mediana_) {}
00091
00096     // Copy assignment
00097     Studentas& operator = (const Studentas& other)
00098     {
00099         if (this != &other) {
00100             Zmogus::setVardas(other.getVardas());
00101             Zmogus::setPavarde(other.getPavarde());
00102             pazymiai_ = other.pazymiai_;
00103             egzamino_rezultatas_ = other.egzamino_rezultatas_;
00104             galutinis_vid_ = other.galutinis_vid_;
00105             mediana_ = other.mediana_;
00106         }
00107         return *this;
00108     }
00109
00114     // Move constructor
00115     Studentas(Studentas&& other) noexcept
```



```

00116         : Zmogus(move(other.vardas_), move(other.pavarde_), pazymiai_(move(other.pazymiai_)),
    egzamino_rezultatas_(other.egzamino_rezultatas_),
00117         galutinis_vid_(other.galutinis_vid_), mediana_(other.mediana_) {other.egzamino_rezultatas_ =
    0; other.galutinis_vid_ = 0;
00118         other.mediana_ = 0; }
00119
00124     // Move assignment
00125     Studentas& operator = (Studentas&& other) noexcept
00126     {
00127         if (this != &other) {
00128             Zmogus::setVardas(move(other.getVardas()));
00129             Zmogus::setPavarde(move(other.getPavarde()));
00130             pazymiai_ = move(other.pazymiai_);
00131             egzamino_rezultatas_ = other.egzamino_rezultatas_;
00132             galutinis_vid_ = other.galutinis_vid_;
00133             mediana_ = other.mediana_;
00134             other.egzamino_rezultatas_ = 0;
00135             other.galutinis_vid_ = 0;
00136             other.mediana_ = 0;
00137         }
00138         return *this;
00139     }
00140
00145     ~Studentas() {pazymiai_.clear();}
00146
00151     void setVardas(const string& vardas) override { Zmogus::setVardas(vardas); }
00152
00157     void setPavarde(const string& pavarde) override { Zmogus::setPavarde(pavarde); }
00158
00163     void setPazymiai(const vector<int>& pazymiai) { pazymiai_ = pazymiai; }
00164
00169     void setEgzaminoRezultatas(int egzamino_rezultatas) { egzamino_rezultatas_ = egzamino_rezultatas;
    }
00170
00175     void setGalutinisVid(double galutinis_vid) { galutinis_vid_ = galutinis_vid; }
00176
00181     void setMediana(double mediana) { mediana_ = mediana; }
00182
00183
00187     string getVardas() const override { return vardas_; }
00188
00192     string getPavarde() const override { return pavarde_; }
00193
00197     const vector<int>& getPazymiai() const { return pazymiai_; }
00198
00202     int getEgzaminoRezultatas() const { return egzamino_rezultatas_; }
00203
00207     double getGalutinisVid() const { return galutinis_vid_; }
00208
00212     double getMediana() const { return mediana_; }
00213
00218     void addPazymys(int pazymys) { pazymiai_.push_back(pazymys); }
00219
00220
00228     friend istream& operator>>(istream& is, Studentas& studentas) {
00229
00230         is >> studentas.vardas_ >> studentas.pavarde_;
00231
00232         int pazymys;
00233         studentas.pazymiai_.clear();
00234         while(is >> pazymys && pazymys > 0 && pazymys <= 10 ) {
00235             studentas.addPazymys(pazymys);
00236         }
00237
00238         is >> studentas.egzamino_rezultatas_;
00239
00240         return is;
00241     }
00242
00250     friend ostream& operator<<(ostream& os, const Studentas& studentas) {
00251         os << "Studento informacija:" << endl;
00252         os << "Vardas: " << studentas.vardas_ << endl;
00253         os << "Pavarde: " << studentas.pavarde_ << endl;
00254         os << "Galutinis vidurkis: " << studentas.galutinis_vid_ << endl;
00255         return os;
00256     }
00257
00258 };
00259
00260 #endif

```

5.6 C:/Users/acer/Desktop/Adri/VSCode/Programavimas/Vektoriai.cpp

File Reference

```
#include <iostream>
#include <cmath>
#include <math.h>
#include <iomanip>
#include <string>
#include <algorithm>
#include <sstream>
#include <cstdlib>
#include <ctime>
#include <vector>
#include <fstream>
#include <chrono>
#include <random>
#include "studentas.h"
#include "funkcijos.h"
```

Functions

- int `main` ()
Pagrindinė programa.

Variables

- int `m`
- int `n`
- int `menu`
- int `variantas_namu_darbas`
- int `variantas_studentas`
- int `pasirinkimas`
- double `suma`
- double `vidurkis`
- int `MAX_ND` = 30
- int `MAX_STUDENTU` = 30
- string `raide`

5.6.1 Function Documentation

5.6.1.1 `main()`

```
int main ( )
```

Pagrindinė programa.

Programa, kurioje naudojas pasirenka ..

5.6.2 Variable Documentation

5.6.2.1 m

```
int m
```

5.6.2.2 MAX_ND

```
int MAX_ND = 30
```

5.6.2.3 MAX_STUDENTU

```
int MAX_STUDENTU = 30
```

5.6.2.4 menu

```
int menu
```

5.6.2.5 n

```
int n
```

5.6.2.6 pasirinkimas

```
int pasirinkimas
```

5.6.2.7 raide

```
string raide
```

5.6.2.8 suma

```
double suma
```

5.6.2.9 variantas_namu_darbas

```
int variantas_namu_darbas
```

5.6.2.10 variantas_studentas

```
int variantas_studentas
```

5.6.2.11 vidurkis

```
double vidurkis
```


Index

- ~Studentas
 - Studentas, [9](#)
- ~Zmogus
 - Zmogus, [14](#)
- addPazymys
 - Studentas, [9](#)
- ar_vargsiukas
 - Funkcijos.cpp, [18](#)
 - funkcijos.h, [22](#)
- C:/Users/acer/Desktop/Adri/VSCode/Programavimas/Funkcijos.cpp, [17](#)
- C:/Users/acer/Desktop/Adri/VSCode/Programavimas/funkcijos.h, [21](#), [25](#)
- C:/Users/acer/Desktop/Adri/VSCode/Programavimas/studentas.h, [25](#), [26](#)
- C:/Users/acer/Desktop/Adri/VSCode/Programavimas/Vektoriai.cpp, [28](#)
- failolsvedimasDeque1
 - Funkcijos.cpp, [18](#)
 - funkcijos.h, [22](#)
- failolsvedimasList1
 - Funkcijos.cpp, [18](#)
 - funkcijos.h, [22](#)
- failolsvedimasVector1
 - Funkcijos.cpp, [18](#)
 - funkcijos.h, [22](#)
- Funkcijos.cpp
 - ar_vargsiukas, [18](#)
 - failolsvedimasDeque1, [18](#)
 - failolsvedimasList1, [18](#)
 - failolsvedimasVector1, [18](#)
 - generuotiFaila, [18](#)
 - mediana, [18](#)
 - nuskaitytiFailaDeque1, [18](#)
 - nuskaitytiFailaDeque2, [19](#)
 - nuskaitytiFailaDeque3, [19](#)
 - nuskaitytiFailaList1, [19](#)
 - nuskaitytiFailaList2, [19](#)
 - nuskaitytiFailaList3, [19](#)
 - nuskaitytiFailaVector1, [19](#)
 - nuskaitytiFailaVector2, [19](#)
 - nuskaitytiFailaVector3, [20](#)
 - palyginti_pagal_galutini_vidurki, [20](#)
 - palyginti_pagal_galutini_vidurki_didejimo_tvarka, [20](#)
 - palyginti_pagal_mediana, [20](#)
 - palyginti_pagal_pavarde, [20](#)
- palyginti_pagal_varda, [20](#)
- RezultatuVaizdavimas, [20](#)
- testai, [20](#)
- vidurkis_galutinis, [21](#)
- funkcijos.h
 - ar_vargsiukas, [22](#)
 - failolsvedimasDeque1, [22](#)
 - failolsvedimasList1, [22](#)
 - failolsvedimasVector1, [22](#)
 - generuotiFaila, [22](#)
 - mediana, [22](#)
 - nuskaitytiFailaDeque1, [22](#)
 - nuskaitytiFailaDeque2, [22](#)
 - nuskaitytiFailaDeque3, [23](#)
 - nuskaitytiFailaList1, [23](#)
 - nuskaitytiFailaList2, [23](#)
 - nuskaitytiFailaList3, [23](#)
 - nuskaitytiFailaVector1, [23](#)
 - nuskaitytiFailaVector2, [23](#)
 - nuskaitytiFailaVector3, [23](#)
 - palyginti_pagal_galutini_vidurki, [24](#)
 - palyginti_pagal_galutini_vidurki_didejimo_tvarka, [24](#)
 - palyginti_pagal_mediana, [24](#)
 - palyginti_pagal_pavarde, [24](#)
 - palyginti_pagal_varda, [24](#)
 - RezultatuVaizdavimas, [24](#)
 - testai, [24](#)
 - vidurkis_galutinis, [24](#)
- generuotiFaila
 - Funkcijos.cpp, [18](#)
 - funkcijos.h, [22](#)
- getEgzaminoRezultatas
 - Studentas, [10](#)
- getGalutinisVid
 - Studentas, [10](#)
- getMediana
 - Studentas, [10](#)
- getPavarde
 - Studentas, [10](#)
 - Zmogus, [15](#)
- getPazymiai
 - Studentas, [10](#)
- getVardas
 - Studentas, [10](#)
 - Zmogus, [15](#)
- m
 - Vektoriai.cpp, [29](#)

main
 Vektoriai.cpp, 28
MAX_ND
 Vektoriai.cpp, 29
MAX_STUDENTU
 Vektoriai.cpp, 29
mediana
 Funkcijos.cpp, 18
 funkcijos.h, 22
menu
 Vektoriai.cpp, 29
n
 Vektoriai.cpp, 29
nuskaitytiFailaDeque1
 Funkcijos.cpp, 18
 funkcijos.h, 22
nuskaitytiFailaDeque2
 Funkcijos.cpp, 19
 funkcijos.h, 22
nuskaitytiFailaDeque3
 Funkcijos.cpp, 19
 funkcijos.h, 23
nuskaitytiFailaList1
 Funkcijos.cpp, 19
 funkcijos.h, 23
nuskaitytiFailaList2
 Funkcijos.cpp, 19
 funkcijos.h, 23
nuskaitytiFailaList3
 Funkcijos.cpp, 19
 funkcijos.h, 23
nuskaitytiFailaVector1
 Funkcijos.cpp, 19
 funkcijos.h, 23
nuskaitytiFailaVector2
 Funkcijos.cpp, 19
 funkcijos.h, 23
nuskaitytiFailaVector3
 Funkcijos.cpp, 20
 funkcijos.h, 23
operator<<
 Studentas, 12
operator>>
 Studentas, 13
operator=
 Studentas, 10, 11
palyginti_pagal_galutini_vidurki
 Funkcijos.cpp, 20
 funkcijos.h, 24
palyginti_pagal_galutini_vidurki_didejimo_tvarka
 Funkcijos.cpp, 20
 funkcijos.h, 24
palyginti_pagal_mediana
 Funkcijos.cpp, 20
 funkcijos.h, 24
palyginti_pagal_pavarde
 Funkcijos.cpp, 20
 funkcijos.h, 24
pasirinkimas
 Vektoriai.cpp, 29
pavarde_
 Zmogus, 16
raide
 Vektoriai.cpp, 29
RezultatuVaizdavimas
 Funkcijos.cpp, 20
 funkcijos.h, 24
setEgzaminoRezultatas
 Studentas, 11
setGalutinisVid
 Studentas, 11
setMediana
 Studentas, 11
setPavarde
 Studentas, 12
 Zmogus, 15
setPazymiai
 Studentas, 12
setVardas
 Studentas, 12
 Zmogus, 15
Studentas, 7
 ~Studentas, 9
 addPazymys, 9
 getEgzaminoRezultatas, 10
 getGalutinisVid, 10
 getMediana, 10
 getPavarde, 10
 getPazymiai, 10
 getVaras, 10
 operator<<, 12
 operator>>, 13
 operator=, 10, 11
 setEgzaminoRezultatas, 11
 setGalutinisVid, 11
 setMediana, 11
 setPavarde, 12
 setPazymiai, 12
 setVardas, 12
 Studentas, 9
suma
 Vektoriai.cpp, 29
testai
 Funkcijos.cpp, 20
 funkcijos.h, 24
vardas_
 Zmogus, 16
variantas_namu_darbas

- Vektoriai.cpp, [29](#)
- variantas_studentas
 - Vektoriai.cpp, [29](#)
- Vektoriai.cpp
 - m, [29](#)
 - main, [28](#)
 - MAX_ND, [29](#)
 - MAX_STUDENTU, [29](#)
 - menu, [29](#)
 - n, [29](#)
 - pasirinkimas, [29](#)
 - raide, [29](#)
 - suma, [29](#)
 - variantas_namu_darbas, [29](#)
 - variantas_studentas, [29](#)
 - vidurkis, [29](#)
- vidurkis
 - Vektoriai.cpp, [29](#)
- vidurkis_galutinis
 - Funkcijos.cpp, [21](#)
 - funkcijos.h, [24](#)
- Zmogus, [13](#)
 - ~Zmogus, [14](#)
 - getPavarde, [15](#)
 - getVardas, [15](#)
 - pavarde_, [16](#)
 - setPavarde, [15](#)
 - setVardas, [15](#)
 - vardas_, [16](#)
 - Zmogus, [14](#)