

OOPuzdAJasiukevicius

Generated by Doxygen 1.10.0



<b>1 Hierarchical Index</b>	<b>1</b>
1.1 Class Hierarchy	1
<b>2 Class Index</b>	<b>3</b>
2.1 Class List	3
<b>3 File Index</b>	<b>5</b>
3.1 File List	5
<b>4 Class Documentation</b>	<b>7</b>
4.1 studentasV Class Reference	7
4.1.1 Constructor & Destructor Documentation	8
4.1.1.1 studentasV() [1/3]	8
4.1.1.2 studentasV() [2/3]	8
4.1.1.3 studentasV() [3/3]	9
4.1.2 Member Function Documentation	9
4.1.2.1 getEgzaminas()	9
4.1.2.2 getMediana()	9
4.1.2.3 getPavarde()	9
4.1.2.4 getPazymiai()	10
4.1.2.5 getVardas()	10
4.1.2.6 getVidurkis()	10
4.1.2.7 operator=() [1/2]	10
4.1.2.8 operator=() [2/2]	10
4.1.2.9 resizePazymiai()	11
4.1.2.10 setAtsitiktiniaiDuomenys()	11
4.1.2.11 setEgzaminas()	11
4.1.2.12 setPavarde()	11
4.1.2.13 setPazymiai()	12
4.1.2.14 setPazymiaiVector()	12
4.1.2.15 setVardas()	12
4.1.3 Friends And Related Symbol Documentation	12
4.1.3.1 operator<<	12
4.1.3.2 operator>>	13
4.2 Zmogus Class Reference	13
4.2.1 Detailed Description	14
4.2.2 Member Function Documentation	14
4.2.2.1 getPavarde()	14
4.2.2.2 getVardas()	14
4.2.2.3 setPavarde()	14
4.2.2.4 setVardas()	15
<b>5 File Documentation</b>	<b>17</b>
5.1 Common.h	17

5.2 PazymiaiVectors.h . . . . .	17
5.3 studentas.h . . . . .	18
<b>Index</b>	<b>21</b>

# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

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

Zmogus . . . . .	<a href="#">13</a>
studentasV . . . . .	<a href="#">7</a>



## Chapter 2

# Class Index

### 2.1 Class List

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

<a href="#">studentasV</a> . . . . .	<a href="#">7</a>
<a href="#">Zmogus</a>	
Base klasė <a href="#">Zmogus</a> kuria zmogaus objektus - vardas, pavarde . . . . .	<a href="#">13</a>





## Chapter 3

# File Index

### 3.1 File List

Here is a list of all documented files with brief descriptions:

<a href="#">Common.h</a>	17
<a href="#">PazymiaiVectors.h</a>	17
<a href="#">studentas.h</a>	18

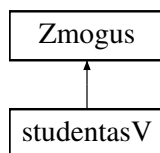


## Chapter 4

# Class Documentation

### 4.1 studentasV Class Reference

Inheritance diagram for studentasV:



#### Public Member Functions

- `studentasV ()`  
*Default studento objekto konstruktorius.*
- void `setVardas` (std::string vardas) override  
*Override'inta `Zmogus` klases funkcija `setVardas` - nustato studento varda.*
- std::string `getVardas` () const override  
*Override'inta `Zmogus` klases funkcija `getVardas` - grazina studento varda.*
- void `setPavarde` (std::string pavarde) override  
*Override'inta `Zmogus` klases funkcija `setPavarde` - nustato studento pavarde.*
- std::string `getPavarde` () const override  
*Override'inta `Zmogus` klases funkcija `getPavarde` - grazina studento pavarde.*
- void `setPazymiaiVector` (std::vector< int > pazVector)  
*funkcija `setPazymiaiVector` - priskiriami pazymiai (vector<int> tipo).*
- void `setPazymiai` (int paz)  
*funkcija `setPazymiai` - priskiriami pazymiai (int tipo).*
- std::vector< int > `getPazymiai` () const  
*funkcija `getPazymiai` - grazina pazymius.*
- void `resizePazymiai` (int n)  
*funkcija `resizePazymiai` - pakeicia std::vector<int> pazymiai dydi.*
- void `setEgzaminas` (int egzaminas)  
*funkcija `setEgzaminas` - iraso studento egzamino rezultata.*
- int `getEgzaminas` () const  
*funkcija `getEgzaminas` - grazina mokinio egzamino rezultata.*

- void **setVidurkis** ()  
*funkcija setVidurkis - apskaiciuoja studento vidurki.*
- void **setMediana** ()  
*funkcija setMediana - apskaiciuoja studento mediana.*
- float **getVidurkis** () const  
*funkcija getVidurkis - grazina studento vidurki.*
- float **getMediana** () const  
*funkcija getMediana - grazina studento mediana.*
- void **setAtsitiktiniaiPazymiai** ()  
*funkcija setAtsitiktiniaiPazymiai - ivestiems studentu vardams ranka, sugeneruoja pazymius ir egzamina.*
- void **setAtsitiktiniaiDuomenys** ()
- **studentasV** (const **studentasV** &kita)  
*rule of five kopijavimo konstruktorius - kopijuoja visa objekta su jo duomenimis i kita objekta.*
- **studentasV** & **operator=** (const **studentasV** &kita)  
*rule of five priskyrimo konstruktorius - priskiria vieno objekto duomenis kitam objektui*
- **studentasV** (**studentasV** &&kita) noexcept  
*rule of five perkavimo konstruktorius - perkeli objekta is vieno i kita pasalindamas duomenis is pirmojo*
- **studentasV** & **operator=** (**studentasV** &&kita) noexcept  
*rule of five perkavimo priskyrimo operatorius - perkeli objekta is vieno i kita pasalindamas duomenis is pirmojo*

## Public Member Functions inherited from **Zmogus**

- virtual ~**Zmogus** ()=default  
*Virtualus objekto destruktorius.*

## Friends

- std::ostream & **operator<<** (std::ostream &os, const **studentasV** &studentas)  
*Overloadinamas ostream operatorius << skirtas darbui su objektu studentas.*
- std::istream & **operator>>** (std::istream &is, **studentasV** &studentas)  
*Overloadinamas istream operatorius >> skirtas darbui su objektu studentas.*

## 4.1.1 Constructor & Destructor Documentation

### 4.1.1.1 **studentasV**() [1/3]

```
studentasV::studentasV ( ) [inline]
```

Default studento objekto konstruktorius.

>Studento egzamino ivertinimas.

### 4.1.1.2 **studentasV**() [2/3]

```
studentasV::studentasV (
    const studentasV & kita )
```

rule of five kopijavimo konstruktorius - kopijuoja visa objekta su jo duomenimis i kita objekta.

## Parameters

<i>kita</i>	
-------------	--

**4.1.1.3 studentasV()** [3/3]

```
studentasV::studentasV (
    studentasV && kita )    [noexcept]
```

rule of five perkeliimo konstruktorius - perkelia objekta is vieno i kita pasalindamas duomenis is pirmojo

## Parameters

<i>kita</i>	
-------------	--

**4.1.2 Member Function Documentation****4.1.2.1 getEgzaminas()**

```
int studentasV::getEgzaminas ( ) const
```

funkcija getEgzaminas - grazina mokinio egzamino rezultata.

## Returns

egzaminas

**4.1.2.2 getMediana()**

```
float studentasV::getMediana ( ) const
```

funkcija getMediana - grazina studento mediana.

## Returns

mediana

**4.1.2.3 getPavarde()**

```
std::string studentasV::getPavarde ( ) const    [override], [virtual]
```

Override'inta [Zmogus](#) klases funkcija getPavarde - grazina studento pavarde.

## Returns

pavarde

Implements [Zmogus](#).

#### 4.1.2.4 getPazymiai()

```
std::vector< int > studentasV::getPazymiai ( ) const
```

funkcija getPazymiai - grazina pazymius.

##### Returns

std::vector<int> pazymiai

#### 4.1.2.5 getVardas()

```
std::string studentasV::getVardas ( ) const [override], [virtual]
```

Override'inta [Zmogus](#) klases funkcija getVardas - grazina studento varda.

##### Returns

vardas

Implements [Zmogus](#).

#### 4.1.2.6 getVidurkis()

```
float studentasV::getVidurkis ( ) const
```

funkcija getVidurkis - grazina studento vidurki.

##### Returns

vidurkis

#### 4.1.2.7 operator=() [1/2]

```
studentasV & studentasV::operator= (
    const studentasV & kita )
```

rule of five priskyrimo konstruktorius - priskiria vieno objekto duomenis kitam objektui

##### Parameters

<i>kita</i>	
-------------	--

#### 4.1.2.8 operator=() [2/2]

```
studentasV & studentasV::operator= (
    studentasV && kita ) [noexcept]
```

rule of five perkeliimo priskyrimo operatorius - perkelia objekta is vieno i kita pasalindamas duomenis is pirmojo

#### Parameters

<i>kita</i>	
-------------	--

#### 4.1.2.9 `resizePazymiai()`

```
void studentasV::resizePazymiai (
    int n )
```

funkcija `resizePazymiai` - pakeicia `std::vector<int>` pazymiai dydi.

#### Parameters

<i>n</i>	
----------	--

#### 4.1.2.10 `setAtsitiktiniaiDuomenys()`

```
void studentasV::setAtsitiktiniaiDuomenys ( )
```

funkcija `setAtsitiktiniaiDuomenys` - sugeneruoja studentus ir ju pazymius ir egzamina.

#### 4.1.2.11 `setEgzaminas()`

```
void studentasV::setEgzaminas (
    int egzaminas )
```

funkcija `setEgzaminas` - iraso studento egzamino rezultata.

#### Parameters

<i>egzaminas</i>	
------------------	--

#### 4.1.2.12 `setPavarde()`

```
void studentasV::setPavarde (
    std::string pavarde ) [override], [virtual]
```

Override'inta [Zmogus](#) klases funkcija `setPavarde` - nustato studento pavarde.

#### Parameters

<i>pavarde</i>	
----------------	--

Implements [Zmogus](#).

#### 4.1.2.13 setPazymiai()

```
void studentasV::setPazymiai (
    int paz )
```

funkcija setPazymiai - priskiriami pazymiai (int tipo).

##### Parameters

<i>paz</i>	
------------	--

#### 4.1.2.14 setPazymiaiVector()

```
void studentasV::setPazymiaiVector (
    std::vector< int > pazVector )
```

funkcija setPazymiaiVector - priskiriami pazymiai (vector<int> tipo).

##### Parameters

<i>pazVector</i>	
------------------	--

#### 4.1.2.15 setVardas()

```
void studentasV::setVardas (
    std::string vardas ) [override], [virtual]
```

Override'inta [Zmogus](#) klases funkcija setVardas - nustato studento varda.

##### Parameters

<i>vardas</i>	
---------------	--

Implements [Zmogus](#).

### 4.1.3 Friends And Related Symbol Documentation

#### 4.1.3.1 operator<<

```
std::ostream & operator<< (
    std::ostream & os,
    const studentasV & studentas ) [friend]
```

Overloadinamas ostream operatorius << skirtas darbui su objektu studentas.



## Parameters

<i>os</i>	outputstream kintamasis
<i>studentas</i>	

## 4.1.3.2 operator&gt;&gt;

```
std::istream & operator>> (
    std::istream & is,
    studentasV & studentas ) [friend]
```

Overloadinamas istream operatorius >> skirtas darbui su objektu studentas.

## Parameters

<i>is</i>	
<i>studentas</i>	

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

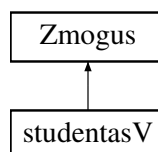
- studentas.h
- studentas.cpp

## 4.2 Zmogus Class Reference

Base klasė [Zmogus](#) kuria zmogaus objektus - vardas, pavarde.

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

Inheritance diagram for Zmogus:



## Public Member Functions

- virtual void [setVardas](#) (std::string vardas)=0  
*Objektui priskiria vardą.*
- virtual std::string [getVardas](#) () const =0  
*Grazina objekto vardą.*
- virtual void [setPavarde](#) (std::string pavarde)=0  
*Objektui priskiria pavardę.*
- virtual std::string [getPavarde](#) () const =0  
*Grazina objekto pavardę.*
- virtual ~**Zmogus** ()=default  
*Virtualus objekto destruktorius.*

### 4.2.1 Detailed Description

Base klasė [Zmogus](#) kuria zmogaus objektus - vardas, pavarde.

Ši klasė yra abstrakti, t.y. apibrezia abstrakčius metodus, kurie turi būti perkrauti paveldetose klasėse

### 4.2.2 Member Function Documentation

#### 4.2.2.1 getPavarde()

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

Grazina objekto pavarde.

##### Returns

pavarde Grazinama pavarde.

Implemented in [studentasV](#).

#### 4.2.2.2 getVardas()

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

Grazina objekto varda.

##### Returns

vardas Grazinamas objekto vardas.

Implemented in [studentasV](#).

#### 4.2.2.3 setPavarde()

```
virtual void Zmogus::setPavarde (
    std::string pavarde ) [pure virtual]
```

Objektui priskiria pavarde.

##### Parameters

<i>pavarde</i>	Nustatoma pavarde.
----------------	--------------------

Implemented in [studentasV](#).

#### 4.2.2.4 setVardas()

```
virtual void Zmogus::setVardas (
    std::string vardas ) [pure virtual]
```

Objektui priskiria vardą.

##### Parameters

<i>vardas</i>	Nustatomas vardas.
---------------	--------------------

Implemented in [studentasV](#).

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

- studentas.h



# Chapter 5

## File Documentation

### 5.1 Common.h

```
00001 #pragma once
00002 #include "PazymiaiVectors.h"
00003
00008 void menu(string& pasirinkimas);
00009
00014 void sortChoice(string& choice);
00015
00020 void pasirinkimasVidMed(string& vidMed);
00021
00026 void pasirinkimasIvedimas(string& ivedimas);
00027
00032 void failoGeneravimasIsvedimas(int iteracija);
00033
00037 void pazymiuFailoGeneravimas();
```

### 5.2 PazymiaiVectors.h

```
00001 #pragma once
00002 #include <iostream>
00003 #include <string>
00004 #include <iomanip>
00005 #include <cstring>
00006 #include <utility>
00007 #include <vector>
00008 #include <numeric>
00009 #include "studentas.h"
00010 using namespace std;
00011
00017 void ivedimas(vector<studentasV>& grupeVector, int n);
00018
00023 void ivedimasNoSize(vector<studentasV>& grupeVector);
00024
00029 void ivedimasCaseTwo(vector<studentasV>& grupeVector);
00030
00039 void fileReading(vector<studentasV>& grupeVector, const string& failas, double & laikasSkaitymas, int&
fakePazymiai, double& laikasSkaiciavimas);
00040
00045 void generateRandomGrades(studentasV &stud);
00046
00051 void generateRandomNames(studentasV &stud);
00052
00057 void generalVidurkisCalculate(vector<studentasV>& grupeVector);
00058
00063 void generalMedianaCalculate(vector<studentasV>& grupeVector);
00064
00074 void isvedimas(vector<studentasV> grupeVector, double laikasSkaitymas, double laikasSkaiciavimas,
double laikasRusiavimas, int fakePazymiai, int iteracija);
00075
00081 void readNumbersV(studentasV& stud, int maxItems);
00082
00088 void sortInput(string& choice, vector<studentasV>& grupeVector);
00089
00100 void failoNuskaitymasRusiavimas(vector<studentasV>& grupeVector, vector<studentasV>& grupeBad,
vector<studentasV>& grupeGood, int i, string vidMed, string ivedimasKonteineris, string choice);
```

```

00101
00110 void isvedimasFailai(vector<studentasV>& grupeVector, vector<studentasV>& grupeBad, int i, string&
    vidMed, string& choice);
00111
00116 void pasirinkimasVidMed(string& vidMed);
00117
00127 void vektoriaiMain(string vidMed, string choice, vector<studentasV>& grupeVector, vector<studentasV>&
    grupeBad, vector<studentasV>& grupeGood, string ivedimasKonteineris);
00128
00136 void vectorPartition(string vidMed, vector<studentasV>& grupeVector, vector<studentasV>& grupeGood,
    vector<studentasV>& grupeBad);
00137
00144 void vectorPartition2(string vidMed, vector<studentasV>& grupeVector, vector<studentasV>& grupeBad);

```

## 5.3 studentas.h

```

00001 //
00002 // Created by Adomas on 15/04/2024.
00003 //
00004
00005 #ifndef OOPUZD_STUDENTAS_H
00006 #define OOPUZD_STUDENTAS_H
00007
00008 #include <iostream>
00009 #include <string>
00010 #include <vector>
00011 #include "numeric"
00012 #include <algorithm>
00018 class Zmogus {
00019 public:
00020
00026     virtual void setVardas(std::string vardas) = 0;
00027
00033     virtual std::string getVardas() const = 0;
00034
00040     virtual void setPavarde(std::string pavarde) = 0;
00041
00047     virtual std::string getPavarde() const = 0;
00048
00052     virtual ~Zmogus() = default;
00053 };
00054
00055 class studentasV : public Zmogus {
00056 private:
00057     std::string vardas = " ";
00058     std::string pavarde = " ";
00059     float vidurkis = 0.0;
00060     float mediana = 0.0;
00061     std::vector<int> pazymiai;
00062     double egzaminas{};
00063 public:
00064
00068     studentasV() : egzaminas(0.0) {}
00069
00075     void setVardas(std::string vardas) override;
00076
00082     std::string getVardas() const override;
00083
00089     void setPavarde(std::string pavarde) override;
00090
00096     std::string getPavarde() const override;
00097
00103     void setPazymiaiVector(std::vector<int> pazVector);
00104
00110     void setPazymiai(int paz);
00111
00116     std::vector<int> getPazymiai() const;
00117
00122     void resizePazymiai(int n);
00127     void setEgzaminas(int egzaminas);
00128
00133     int getEgzaminas() const;
00134
00138     void setVidurkis();
00139
00143     void setMediana();
00144
00149     float getVidurkis() const;
00150
00155     float getMediana() const;
00156
00160     void setAtsitiktiniaiPazymiai();
00161

```

```
00165     void setAtsitiktiniaiDuomenys();
00166
00167 //     //Rule of Five headers
00172     studentasV(const studentasV &kita);
00173
00178     studentasV &operator=(const studentasV &kita);
00179
00184     studentasV(studentasV &&kita) noexcept;
00185
00190     studentasV &operator=(studentasV &&kita) noexcept; //Perkelimo priskyrimo konstruktorius
00191     ~studentasV(); //Destruktorius
00192
00198     friend std::ostream &operator<<(std::ostream &os, const studentasV &studentas);
00199
00205     friend std::istream &operator>>(std::istream &is, studentasV &studentas);
00206 };
00207
00208 #endif //OOPUZD_STUDENTAS_H
```





# Index

- getEgzaminas
  - studentasV, [9](#)
- getMediana
  - studentasV, [9](#)
- getPavarde
  - studentasV, [9](#)
  - Zmogus, [14](#)
- getPazymiai
  - studentasV, [9](#)
- getVardas
  - studentasV, [10](#)
  - Zmogus, [14](#)
- getVidurkis
  - studentasV, [10](#)
  
- operator<<
  - studentasV, [12](#)
- operator>>
  - studentasV, [13](#)
- operator=
  - studentasV, [10](#)
  
- resizePazymiai
  - studentasV, [11](#)
  
- setAtsitiktiniaiDuomenys
  - studentasV, [11](#)
- setEgzaminas
  - studentasV, [11](#)
- setPavarde
  - studentasV, [11](#)
  - Zmogus, [14](#)
- setPazymiai
  - studentasV, [11](#)
- setPazymiaiVector
  - studentasV, [12](#)
- setVardas
  - studentasV, [12](#)
  - Zmogus, [14](#)
- studentasV, [7](#)
  - getEgzaminas, [9](#)
  - getMediana, [9](#)
  - getPavarde, [9](#)
  - getPazymiai, [9](#)
  - getVardas, [10](#)
  - getVidurkis, [10](#)
  - operator<<, [12](#)
  - operator>>, [13](#)
  - operator=, [10](#)
  - resizePazymiai, [11](#)
  - setAtsitiktiniaiDuomenys, [11](#)
  - setEgzaminas, [11](#)
  - setPavarde, [11](#)
  - setPazymiai, [11](#)
  - setPazymiaiVector, [12](#)
  - setVardas, [12](#)
  - studentasV, [8](#), [9](#)
- Zmogus, [13](#)
  - getPavarde, [14](#)
  - getVardas, [14](#)
  - setPavarde, [14](#)
  - setVardas, [14](#)