

obj

Generated by Doxygen 1.12.0



<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 Studentas Class Reference	7
4.1.1 Constructor & Destructor Documentation	8
4.1.1.1 Studentas() [1/3]	8
4.1.1.2 Studentas() [2/3]	8
4.1.1.3 Studentas() [3/3]	8
4.1.1.4 ~Studentas()	8
4.1.2 Member Function Documentation	9
4.1.2.1 galutineMediana()	9
4.1.2.2 galutinisVidurkis()	9
4.1.2.3 generuotiStudentuDuomenis()	9
4.1.2.4 getEgzaminas()	9
4.1.2.5 getGalutinis()	9
4.1.2.6 getNamuDarbai()	9
4.1.2.7 isvestiInformacija()	9
4.1.2.8 nuskaitytiFailo()	9
4.1.2.9 operator=()	10
4.1.2.10 rasytiFaila()	10
4.1.2.11 setEgzaminas()	10
4.1.2.12 setGalutinis()	10
4.1.2.13 setNamuDarbai()	10
4.1.3 Friends And Related Symbol Documentation	10
4.1.3.1 operator<<	10
4.1.3.2 operator>>	10
4.2 Zmogus Class Reference	11
4.2.1 Constructor & Destructor Documentation	11
4.2.1.1 Zmogus() [1/2]	11
4.2.1.2 Zmogus() [2/2]	11
4.2.1.3 ~Zmogus()	11
4.2.2 Member Function Documentation	11
4.2.2.1 getPavarde()	11
4.2.2.2 getVardas()	12
4.2.2.3 isvestiInformacija()	12
4.2.3 Member Data Documentation	12

4.2.3.1 pavarde_ . . . . .	12
4.2.3.2 vardas_ . . . . .	12
<b>5 File Documentation</b>	<b>13</b>
5.1 failai.cpp File Reference . . . . .	13
5.1.1 Function Documentation . . . . .	13
5.1.1.1 atmintis() . . . . .	13
5.1.1.2 generuotiStudentuDuomenis() . . . . .	13
5.1.1.3 matuotiVeikimoGreiti() . . . . .	14
5.1.1.4 nuskaitytiStudentus() . . . . .	14
5.1.1.5 palyginimas_pavarde() . . . . .	14
5.1.1.6 palyginimas_pazimys() . . . . .	14
5.1.1.7 palyginimas_vardas() . . . . .	14
5.1.1.8 rusiavimas_pavarde() . . . . .	14
5.1.1.9 rusiavimas_pazimys() . . . . .	14
5.1.1.10 rusiavimas_vardas() . . . . .	15
5.1.1.11 rusiuotiStudentus() . . . . .	15
5.1.1.12 surusioti_failai() . . . . .	15
5.2 failai.h File Reference . . . . .	15
5.2.1 Function Documentation . . . . .	15
5.2.1.1 atmintis() . . . . .	15
5.2.1.2 generuotiStudentuDuomenis() . . . . .	16
5.2.1.3 matuotiVeikimoGreiti() . . . . .	16
5.2.1.4 palyginimas_pavarde() . . . . .	16
5.2.1.5 palyginimas_pazimys() . . . . .	16
5.2.1.6 palyginimas_vardas() . . . . .	16
5.2.1.7 rusiavimas_pavarde() . . . . .	16
5.2.1.8 rusiavimas_pazimys() . . . . .	16
5.2.1.9 rusiavimas_vardas() . . . . .	16
5.2.1.10 rusiuotiStudentus() . . . . .	17
5.2.1.11 surusioti_failai() . . . . .	17
5.3 failai.h . . . . .	17
5.4 idk.cpp File Reference . . . . .	17
5.4.1 Function Documentation . . . . .	18
5.4.1.1 main() . . . . .	18
5.5 list.cpp File Reference . . . . .	18
5.5.1 Function Documentation . . . . .	18
5.5.1.1 atmintisList() . . . . .	18
5.5.1.2 isvestisList() . . . . .	18
5.5.1.3 matuotiVeikimoGreitiList() . . . . .	18
5.5.1.4 nuskaitytilsFailoList() . . . . .	19
5.5.1.5 rusiavimas_pavardeList() . . . . .	19

5.5.1.6 rusiavimas_pazimysList()	19
5.5.1.7 rusiavimas_vardasList()	19
5.5.1.8 rusiavimasList()	19
5.5.1.9 rusiuotiStudentusList()	19
5.5.1.10 surusioti_failaiList()	19
5.6 list.h File Reference	20
5.6.1 Function Documentation	20
5.6.1.1 atmintisList()	20
5.6.1.2 isvestisList()	20
5.6.1.3 matuotiVeikimoGreitiList()	20
5.6.1.4 nuskaitytilsFailoList()	20
5.6.1.5 rusiavimas_pavardeList()	21
5.6.1.6 rusiavimas_pazimysList()	21
5.6.1.7 rusiavimas_vardasList()	21
5.6.1.8 rusiavimasList()	21
5.6.1.9 rusiuotiStudentusList()	21
5.6.1.10 surusioti_failaiList()	21
5.7 list.h	21
5.8 mylib.h File Reference	22
5.9 mylib.h	22
5.10 stud.cpp File Reference	22
5.10.1 Function Documentation	23
5.10.1.1 generuotiAtsitiktiniusRezultatus()	23
5.10.1.2 isvestis()	23
5.10.1.3 nuskaitytilsFailo()	23
5.10.1.4 palyginimas()	23
5.10.1.5 rusiavimas()	23
5.11 stud.h File Reference	23
5.11.1 Function Documentation	24
5.11.1.1 generuotiAtsitiktiniusRezultatus()	24
5.11.1.2 isvestis()	24
5.11.1.3 nuskaitytilsFailo()	24
5.11.1.4 palyginimas()	24
5.11.1.5 rusiavimas()	24
5.12 stud.h	24
<b>Index</b>	<b>29</b>



# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

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

Zmogus . . . . .	<a href="#">11</a>
Studentas . . . . .	<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="#">Studentas</a>	.....	<a href="#">7</a>
<a href="#">Zmogus</a>	.....	<a href="#">11</a>



## Chapter 3

# File Index

### 3.1 File List

Here is a list of all files with brief descriptions:

<a href="#">failai.cpp</a>	13
<a href="#">failai.h</a>	15
<a href="#">idk.cpp</a>	17
<a href="#">list.cpp</a>	18
<a href="#">list.h</a>	20
<a href="#">mylib.h</a>	22
<a href="#">stud.cpp</a>	22
<a href="#">stud.h</a>	23



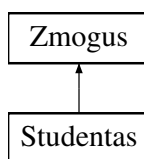
## Chapter 4

# Class Documentation

### 4.1 Studentas Class Reference

```
#include <stud.h>
```

Inheritance diagram for Studentas:



#### Public Member Functions

- [Studentas](#) ()
- [Studentas](#) (const string &vardas, const string &pavarde, const vector< int > &namu\_darbai, int egzaminas)
- [Studentas](#) (const [Studentas](#) &other)
- [Studentas](#) & [operator=](#) (const [Studentas](#) &other)
- [~Studentas](#) ()
- const vector< int > & [getNamuDarbai](#) () const
- int [getEgzaminas](#) () const
- double [getGalutinis](#) () const
- void [setGalutinis](#) (double g)
- void [setNamuDarbai](#) (const vector< int > &nd)
- void [setEgzaminas](#) (int egzaminas)
- void [isvestiInformacija](#) () const override

#### Public Member Functions inherited from [Zmogus](#)

- [Zmogus](#) ()
- [Zmogus](#) (const string &vardas, const string &pavarde)
- virtual [~Zmogus](#) ()
- string [getVardas](#) () const
- string [getPavarde](#) () const

## Static Public Member Functions

- static double [galutinisVidurkis](#) (const vector< int > &namu\_darbai, int egzaminas)
- static double [galutineMediana](#) (vector< int > namu\_darbai, int egzaminas)
- static [Studentas](#) [generuotiStudentuDuomenis](#) ()
- static vector< [Studentas](#) > [nuskaitytilsFailo](#) (const string &failoVardas)
- static void [rasytilFaila](#) (const vector< [Studentas](#) > &studentai, const string &failoVardas)

## Friends

- ostream & [operator<<](#) (ostream &out, const [Studentas](#) &s)
- istream & [operator>>](#) (istream &in, [Studentas](#) &s)

## Additional Inherited Members

## Protected Attributes inherited from [Zmogus](#)

- string [vardas\\_](#)
- string [pavarde\\_](#)

## 4.1.1 Constructor & Destructor Documentation

### 4.1.1.1 [Studentas\(\)](#) [1/3]

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

### 4.1.1.2 [Studentas\(\)](#) [2/3]

```
Studentas::Studentas (
    const string & vardas,
    const string & pavarde,
    const vector< int > & namu_darbai,
    int egzaminas) [inline]
```

### 4.1.1.3 [Studentas\(\)](#) [3/3]

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

### 4.1.1.4 [~Studentas\(\)](#)

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

## 4.1.2 Member Function Documentation

### 4.1.2.1 galutineMediana()

```
double Studentas::galutineMediana (
    vector< int > namu_darbai,
    int egzaminas) [static]
```

### 4.1.2.2 galutinisVidurkis()

```
double Studentas::galutinisVidurkis (
    const vector< int > & namu_darbai,
    int egzaminas) [static]
```

### 4.1.2.3 generuotiStudentuDuomenis()

```
static Studentas Studentas::generuotiStudentuDuomenis () [inline], [static]
```

### 4.1.2.4 getEgzaminas()

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

### 4.1.2.5 getGalutinis()

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

### 4.1.2.6 getNamuDarbai()

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

### 4.1.2.7 isvestiInformacija()

```
void Studentas::isvestiInformacija () const [inline], [override], [virtual]
```

Implements [Zmogus](#).

### 4.1.2.8 nuskaitytiIsFailo()

```
static vector< Studentas > Studentas::nuskaitytiIsFailo (
    const string & failoVardas) [inline], [static]
```

#### 4.1.2.9 operator=()

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

#### 4.1.2.10 rasytiIFaila()

```
static void Studentas::rasytiIFaila (
    const vector< Studentas > & studentai,
    const string & failoVardas) [inline], [static]
```

#### 4.1.2.11 setEgzaminas()

```
void Studentas::setEgzaminas (
    int egzaminas) [inline]
```

#### 4.1.2.12 setGalutinis()

```
void Studentas::setGalutinis (
    double g) [inline]
```

#### 4.1.2.13 setNamuDarbai()

```
void Studentas::setNamuDarbai (
    const vector< int > & nd) [inline]
```

### 4.1.3 Friends And Related Symbol Documentation

#### 4.1.3.1 operator<<

```
ostream & operator<< (
    ostream & out,
    const Studentas & s) [friend]
```

#### 4.1.3.2 operator>>

```
istream & operator>> (
    istream & in,
    Studentas & s) [friend]
```

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

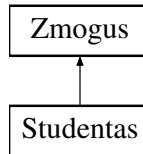
- [stud.h](#)
- [stud.cpp](#)



## 4.2 Zmogus Class Reference

```
#include <stud.h>
```

Inheritance diagram for Zmogus:



### Public Member Functions

- [Zmogus](#) ()
- [Zmogus](#) (const string &vardas, const string &pavarde)
- virtual [~Zmogus](#) ()
- string [getVardas](#) () const
- string [getPavarde](#) () const
- virtual void [isvestiInformacija](#) () const =0

### Protected Attributes

- string [vardas\\_](#)
- string [pavarde\\_](#)

## 4.2.1 Constructor & Destructor Documentation

### 4.2.1.1 Zmogus() [1/2]

```
Zmogus::Zmogus () [inline]
```

### 4.2.1.2 Zmogus() [2/2]

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

### 4.2.1.3 ~Zmogus()

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

## 4.2.2 Member Function Documentation

### 4.2.2.1 getPavarde()

```
string Zmogus::getPavarde () const [inline]
```

#### 4.2.2.2 getVardas()

```
string Zmogus::getVardas () const [inline]
```

#### 4.2.2.3 isvestiInformacija()

```
virtual void Zmogus::isvestiInformacija () const [pure virtual]
```

Implemented in [Studentas](#).

### 4.2.3 Member Data Documentation

#### 4.2.3.1 pavarde\_

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

#### 4.2.3.2 vardas\_

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

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

- [stud.h](#)

# Chapter 5

## File Documentation

### 5.1 failai.cpp File Reference

```
#include "stud.h"
#include "failai.h"
```

#### Functions

- void [atmintis](#) (const vector< [Studentas](#) > &v, const string &name)
- void [generuotiStudentuDuomenis](#) (const string &failoPavadinimas, int kiekis)
- void [nuskaitytiStudentus](#) (const string &failoPavadinimas, vector< [Studentas](#) > &studentai, int pasirinkimas)
- void [rusiuotiStudentus](#) (vector< [Studentas](#) > &studentai, int pasirinkimas, int rusiuotiPagal, int strategija, vector< [Studentas](#) > &vargsiukai, vector< [Studentas](#) > &kietiakiai)
- void [surusioti\\_failai](#) (vector< [Studentas](#) > &studentai, const string &failoPavadinimas)
- bool [palyginimas\\_vardas](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- void [rusiavimas\\_vardas](#) (vector< [Studentas](#) > &studentai)
- bool [palyginimas\\_pavarde](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- void [rusiavimas\\_pavarde](#) (vector< [Studentas](#) > &studentai)
- bool [palyginimas\\_pazimys](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- void [rusiavimas\\_pazimys](#) (vector< [Studentas](#) > &studentai)
- void [matuotiVeikimoGreiti](#) (const string &failoPavadinimas, int kiekis, int pasirinkimas, int rusiuotiPagal, int strategija)

#### 5.1.1 Function Documentation

##### 5.1.1.1 atmintis()

```
void atmintis (
    const vector< Studentas > & v,
    const string & name)
```

##### 5.1.1.2 generuotiStudentuDuomenis()

```
void generuotiStudentuDuomenis (
    const string & failoPavadinimas,
    int kiekis)
```

#### 5.1.1.3 matuotiVeikimoGreiti()

```
void matuotiVeikimoGreiti (
    const string & failoPavadinimas,
    int kiekis,
    int pasirinkimas,
    int rusiuotiPagal,
    int strategija)
```

#### 5.1.1.4 nuskaitytiStudentus()

```
void nuskaitytiStudentus (
    const string & failoPavadinimas,
    vector< Studentas > & studentai,
    int pasirinkimas)
```

#### 5.1.1.5 palyginimas\_pavarde()

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

#### 5.1.1.6 palyginimas\_pazimys()

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

#### 5.1.1.7 palyginimas\_vardas()

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

#### 5.1.1.8 rusiavimas\_pavarde()

```
void rusiavimas_pavarde (
    vector< Studentas > & studentai)
```

#### 5.1.1.9 rusiavimas\_pazimys()

```
void rusiavimas_pazimys (
    vector< Studentas > & studentai)
```

#### 5.1.1.10 rusiavimas\_vardas()

```
void rusiavimas_vardas (
    vector< Studentas > & studentai)
```

#### 5.1.1.11 rusiuotiStudentus()

```
void rusiuotiStudentus (
    vector< Studentas > & studentai,
    int pasirinkimas,
    int rusiuotiPagal,
    int strategija,
    vector< Studentas > & vargsiukai,
    vector< Studentas > & kietiakai)
```

#### 5.1.1.12 surusioti\_failai()

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

## 5.2 failai.h File Reference

```
#include "mylib.h"
#include "stud.h"
```

### Functions

- void [generuotiStudentuDuomenis](#) (const string &failoPavadinimas, int kiekis)
- void [rusiuotiStudentus](#) (vector< [Studentas](#) > &studentai, int pasirinkimas, int rusiuotiPagal, int strategija, vector< [Studentas](#) > &vargsiukai, vector< [Studentas](#) > &kietiakai)
- void [surusioti\\_failai](#) (vector< [Studentas](#) > &studentai, const string &failoPavadinimas)
- bool [palyginimas\\_vardas](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- void [rusiavimas\\_vardas](#) (vector< [Studentas](#) > &studentai)
- bool [palyginimas\\_pavarde](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- void [rusiavimas\\_pavarde](#) (vector< [Studentas](#) > &studentai)
- bool [palyginimas\\_pazimys](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- void [rusiavimas\\_pazimys](#) (vector< [Studentas](#) > &studentai)
- void [matuotiVeikimoGreiti](#) (const string &failoPavadinimas, int kiekis, int pasirinkimas, int rusiuotiPagal, int strategija)
- void [atmintis](#) (const vector< [Studentas](#) > &v, const string &name)

### 5.2.1 Function Documentation

#### 5.2.1.1 atmintis()

```
void atmintis (
    const vector< Studentas > & v,
    const string & name)
```

### 5.2.1.2 generuotiStudentuDuomenis()

```
void generuotiStudentuDuomenis (  
    const string & failoPavadinimas,  
    int kiekis)
```

### 5.2.1.3 matuotiVeikimoGreiti()

```
void matuotiVeikimoGreiti (  
    const string & failoPavadinimas,  
    int kiekis,  
    int pasirinkimas,  
    int rusiuotiPagal,  
    int strategija)
```

### 5.2.1.4 palyginimas\_pavarde()

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

### 5.2.1.5 palyginimas\_pazimys()

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

### 5.2.1.6 palyginimas\_vardas()

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

### 5.2.1.7 rusiavimas\_pavarde()

```
void rusiavimas_pavarde (  
    vector< Studentas > & studentai)
```

### 5.2.1.8 rusiavimas\_pazimys()

```
void rusiavimas_pazimys (  
    vector< Studentas > & studentai)
```

### 5.2.1.9 rusiavimas\_vardas()

```
void rusiavimas_vardas (  
    vector< Studentas > & studentai)
```

### 5.2.1.10 rusiuotiStudentus()

```
void rusiuotiStudentus (
    vector< Studentas > & studentai,
    int pasirinkimas,
    int rusiuotiPagal,
    int strategija,
    vector< Studentas > & vargsiukai,
    vector< Studentas > & kietiakai)
```

### 5.2.1.11 surusioti\_failai()

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

## 5.3 failai.h

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

```
00001 #ifndef FAILAI_H_INCLUDED
00002 #define FAILAI_H_INCLUDED
00003
00004 #include "mylib.h"
00005 #include "stud.h"
00006
00007
00008 void generuotiStudentuDuomenis(const string& failoPavadinimas, int kiekis);
00009 void rusiuotiStudentus(vector<Studentas>& studentai, int pasirinkimas, int rusiuotiPagal, int
    strategija,
00010     vector<Studentas>& vargsiukai, vector<Studentas>& kietiakai);
00011 void surusioti_failai(vector<Studentas>& studentai, const string& failoPavadinimas);
00012 bool palyginimas_vardas(const Studentas& a, const Studentas& b);
00013 void rusiavimas_vardas(vector<Studentas>& studentai);
00014 bool palyginimas_pavarde(const Studentas& a, const Studentas& b);
00015 void rusiavimas_pavarde(vector<Studentas>& studentai);
00016 bool palyginimas_pazimys(const Studentas& a, const Studentas& b);
00017 void rusiavimas_pazimys(vector<Studentas>& studentai);
00018 void matuotiVeikimoGreiti(const string& failoPavadinimas, int kiekis, int pasirinkimas, int
    rusiuotiPagal, int strategija);
00019 void atmintis(const vector<Studentas>& v, const string& name);
00020
00021
00022
00023
00024 #endif
```

## 5.4 idk.cpp File Reference

```
#include "mylib.h"
#include "stud.h"
#include "failai.h"
#include "list.h"
```

### Functions

- int [main](#) ()

## 5.4.1 Function Documentation

### 5.4.1.1 main()

```
int main ()
```

## 5.5 list.cpp File Reference

```
#include "mylib.h"
#include "stud.h"
#include "failai.h"
#include "list.h"
```

### Functions

- void [atmintisList](#) (const list< [Studentas](#) > &lst, const string &name)
- void [isvestisList](#) (const list< [Studentas](#) > &studentaiList, int pasirinkimas)
- void [nuskaitytiIsFailoList](#) (const string &name, list< [Studentas](#) > &studentaiList, int pasirinkimas)
- void [rusiavimasList](#) (list< [Studentas](#) > &studentaiList)
- void [rusiuotiStudentusList](#) (const string &name, int pasirinkimas, int rusiuotiPagal, int strategija, const string &filePrefix, list< [Studentas](#) > &vargsiukaiList, list< [Studentas](#) > &kietiakaiList)
- void [surusioti\\_failaiList](#) (list< [Studentas](#) > &studentaiList, const string &name)
- void [rusiavimas\\_vardasList](#) (list< [Studentas](#) > &studentaiList)
- void [rusiavimas\\_pavardeList](#) (list< [Studentas](#) > &studentaiList)
- void [rusiavimas\\_pazimysList](#) (list< [Studentas](#) > &studentaiList)
- void [matuotiVeikimoGreitiList](#) (const string &name, int kiekis, int pasirinkimas, int rusiuotiPagal, int strategija)

## 5.5.1 Function Documentation

### 5.5.1.1 atmintisList()

```
void atmintisList (
    const list< Studentas > & lst,
    const string & name)
```

### 5.5.1.2 isvestisList()

```
void isvestisList (
    const list< Studentas > & studentaiList,
    int pasirinkimas)
```

### 5.5.1.3 matuotiVeikimoGreitiList()

```
void matuotiVeikimoGreitiList (
    const string & name,
    int kiekis,
    int pasirinkimas,
    int rusiuotiPagal,
    int strategija)
```



#### 5.5.1.4 nuskaitytiIsFailoList()

```
void nuskaitytiIsFailoList (
    const string & name,
    list< Studentas > & studentaiList,
    int pasirinkimas)
```

#### 5.5.1.5 rusiavimas\_pavardeList()

```
void rusiavimas_pavardeList (
    list< Studentas > & studentaiList)
```

#### 5.5.1.6 rusiavimas\_pazimysList()

```
void rusiavimas_pazimysList (
    list< Studentas > & studentaiList)
```

#### 5.5.1.7 rusiavimas\_vardasList()

```
void rusiavimas_vardasList (
    list< Studentas > & studentaiList)
```

#### 5.5.1.8 rusiavimasList()

```
void rusiavimasList (
    list< Studentas > & studentaiList)
```

#### 5.5.1.9 rusiuotiStudentusList()

```
void rusiuotiStudentusList (
    const string & name,
    int pasirinkimas,
    int rusiuotiPagal,
    int strategija,
    const string & filePrefix,
    list< Studentas > & vargsiukaiList,
    list< Studentas > & kietiakiiaiList)
```

#### 5.5.1.10 surusioti\_failaiList()

```
void surusioti_failaiList (
    list< Studentas > & studentaiList,
    const string & name)
```

## 5.6 list.h File Reference

```
#include "mylib.h"
#include "stud.h"
#include "failai.h"
```

### Functions

- void [isvestisList](#) (const list< [Studentas](#) > &studentaiList, int pasirinkimas)
- void [nuskaitytilsFailoList](#) (const string &name, list< [Studentas](#) > &studentaiList, int pasirinkimas)
- void [rusiavimasList](#) (list< [Studentas](#) > &studentaiList)
- void [rusiuotiStudentusList](#) (const string &failoPavadinimas, int pasirinkimas, int rusiuotiPagal, int strategija, const string &filePrefix, list< [Studentas](#) > &vargsiukaiList, list< [Studentas](#) > &kietiakaiList)
- void [surusioti\\_failaiList](#) (list< [Studentas](#) > &studentaiList, const string &failoPavadinimas)
- void [rusiavimas\\_vardasList](#) (list< [Studentas](#) > &studentaiList)
- void [rusiavimas\\_pavardeList](#) (list< [Studentas](#) > &studentaiList)
- void [rusiavimas\\_pazimysList](#) (list< [Studentas](#) > &studentaiList)
- void [matuotiVeikimoGreitiList](#) (const string &failoPavadinimas, int kiekis, int pasirinkimas, int rusiuotiPagal, int strategija)
- void [atmintisList](#) (const std::list< [Studentas](#) > &lst, const string &name)

### 5.6.1 Function Documentation

#### 5.6.1.1 atmintisList()

```
void atmintisList (
    const std::list< Studentas > & lst,
    const string & name)
```

#### 5.6.1.2 isvestisList()

```
void isvestisList (
    const list< Studentas > & studentaiList,
    int pasirinkimas)
```

#### 5.6.1.3 matuotiVeikimoGreitiList()

```
void matuotiVeikimoGreitiList (
    const string & failoPavadinimas,
    int kiekis,
    int pasirinkimas,
    int rusiuotiPagal,
    int strategija)
```

#### 5.6.1.4 nuskaitytilsFailoList()

```
void nuskaitytiIsFailoList (
    const string & name,
    list< Studentas > & studentaiList,
    int pasirinkimas)
```

**5.6.1.5 rusiavimas\_pavardeList()**

```
void rusiavimas_pavardeList (
    list< Studentas > & studentaiList)
```

**5.6.1.6 rusiavimas\_pazimysList()**

```
void rusiavimas_pazimysList (
    list< Studentas > & studentaiList)
```

**5.6.1.7 rusiavimas\_vardasList()**

```
void rusiavimas_vardasList (
    list< Studentas > & studentaiList)
```

**5.6.1.8 rusiavimasList()**

```
void rusiavimasList (
    list< Studentas > & studentaiList)
```

**5.6.1.9 rusiuotiStudentusList()**

```
void rusiuotiStudentusList (
    const string & failoPavadinimas,
    int pasirinkimas,
    int rusiuotiPagal,
    int strategija,
    const string & filePrefix,
    list< Studentas > & vargsiukaiList,
    list< Studentas > & kietiakiiaiList)
```

**5.6.1.10 surusioti\_failaiList()**

```
void surusioti_failaiList (
    list< Studentas > & studentaiList,
    const string & failoPavadinimas)
```

**5.7 list.h**

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

```
00001 #ifndef LIST_H_INCLUDED
00002 #define LIST_H_INCLUDED
00003
00004 #include "mylib.h"
00005 #include "stud.h"
00006 #include "failai.h"
00007
00008 void isvestisList(const list<Studentas>& studentaiList, int pasirinkimas);
00009 void nuskaitytiIsFailoList(const string& name, list<Studentas>& studentaiList, int pasirinkimas);
00010 void rusiavimasList(list<Studentas>& studentaiList);
00011 void rusiuotiStudentusList(const string& failoPavadinimas, int pasirinkimas, int rusiuotiPagal, int
    strategija, const string& filePrefix,
00012     list<Studentas>& vargsiukaiList, list<Studentas>& kietiakiiaiList);
00013 void surusioti_failaiList(list<Studentas>& studentaiList, const string& failoPavadinimas);
00014 void rusiavimas_vardasList(list<Studentas>& studentaiList);
00015 void rusiavimas_pavardeList(list<Studentas>& studentaiList);
00016 void rusiavimas_pazimysList(list<Studentas>& studentaiList);
00017 void matuotiVeikimoGreitiList(const string& failoPavadinimas, int kiekis, int pasirinkimas, int
    rusiuotiPagal, int strategija);
00018 void atmintisList(const std::list<Studentas>& lst, const string& name);
00019
00020 #endif
```

## 5.8 mylib.h File Reference

```
#include <iostream>
#include <iomanip>
#include <vector>
#include <string>
#include <fstream>
#include <numeric>
#include <algorithm>
#include <limits>
#include <stdexcept>
#include <chrono>
#include <random>
#include <list>
#include <sstream>
#include <cstdio>
#include <memory>
#include <array>
```

## 5.9 mylib.h

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

```
00001 #ifndef MYLIB_H
00002 #define MYLIB_H
00003
00004 #include <iostream>
00005 #include <iomanip>
00006 #include <vector>
00007 #include <string>
00008 #include <fstream>
00009 #include <numeric>
00010 #include <algorithm>
00011 #include <limits>
00012 #include <stdexcept>
00013 #include <chrono>
00014 #include <random>
00015 #include <list>
00016 #include <sstream>
00017 #include <cstdio>
00018 #include <memory>
00019 #include <array>
00020
00021 using namespace std;
00022 using namespace std::chrono;
00023
00024 #endif // MYLIB_H
```

## 5.10 stud.cpp File Reference

```
#include "stud.h"
#include "mylib.h"
```

### Functions

- void [isvestis](#) (const vector< [Studentas](#) > &studentai, int pasirinkimas)
- void [generuotiAtsitiktiniusRezultatus](#) ([Studentas](#) &s)
- void [nuskaitytiIsFailo](#) (const string &failoPavadinimas, vector< [Studentas](#) > &studentai)
- bool [palyginimas](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- void [rusiavimas](#) (vector< [Studentas](#) > &studentai)

## 5.10.1 Function Documentation

### 5.10.1.1 generuotiAtsitiktiniusRezultatus()

```
void generuotiAtsitiktiniusRezultatus (  
    Studentas & s)
```

### 5.10.1.2 isvestis()

```
void isvestis (  
    const vector< Studentas > & studentai,  
    int pasirinkimas)
```

### 5.10.1.3 nuskaitytiIsFailo()

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

### 5.10.1.4 palyginimas()

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

### 5.10.1.5 rusiavimas()

```
void rusiavimas (  
    vector< Studentas > & studentai)
```

## 5.11 stud.h File Reference

```
#include "mylib.h"
```

### Classes

- class [Zmogus](#)
- class [Studentas](#)

### Functions

- void [isvestis](#) (const vector< [Studentas](#) > &studentai, int pasirinkimas)
- void [nuskaitytiIsFailo](#) (const string &failoPavadinimas, vector< [Studentas](#) > &studentai)
- void [generuotiAtsitiktiniusRezultatus](#) ([Studentas](#) &s)
- bool [palyginimas](#) (const [Studentas](#) &a, const [Studentas](#) &b)
- void [rusiavimas](#) (vector< [Studentas](#) > &studentai)

## 5.11.1 Function Documentation

### 5.11.1.1 generuotiAtsitiktiniusRezultatus()

```
void generuotiAtsitiktiniusRezultatus (
    Studentas & s)
```

### 5.11.1.2 isvestis()

```
void isvestis (
    const vector< Studentas > & studentai,
    int pasirinkimas)
```

### 5.11.1.3 nuskaitytiIsFailo()

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

### 5.11.1.4 palyginimas()

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

### 5.11.1.5 rusiavimas()

```
void rusiavimas (
    vector< Studentas > & studentai)
```

## 5.12 stud.h

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

```
00001 #ifndef STUD_H
00002 #define STUD_H
00003
00004 #include "mylib.h"
00005
00006 class Zmogus {
00007 protected:
00008     string vardas_;
00009     string pavarde_;
00010
00011 public:
00012     Zmogus() : vardas_(""), pavarde_("") {}
00013     Zmogus(const string& vardas, const string& pavarde) : vardas_(vardas), pavarde_(pavarde) {}
00014     virtual ~Zmogus() {}
00015
00016     string getVardas() const { return vardas_; }
00017     string getPavarde() const { return pavarde_; }
00018
00019     virtual void isvestiInformacija() const = 0;
00020 };
00021
00022 class Studentas : public Zmogus {
```

```

00023 private:
00024     vector<int> nd_;
00025     int egzaminas_;
00026     double galutinis_;
00027
00028 public:
00029     Studentas() : Zmogus(), egzaminas_(0), galutinis_(0.0) {}
00030
00031     Studentas(const string& vardas, const string& pavarde, const vector<int>& namu_darbai, int
egzaminas)
00032         : Zmogus(vardas, pavarde), nd_(namu_darbai), egzaminas_(egzaminas) {
00033         galutinis_ = galutinisVidurkis(nd_, egzaminas_);
00034     }
00035
00036
00037     Studentas(const Studentas& other)
00038         : Zmogus(other), nd_(other.nd_), egzaminas_(other.egzaminas_), galutinis_(other.galutinis_) {}
00039
00040     Studentas& operator=(const Studentas& other) {
00041         if (this == &other) return *this;
00042         vardas_ = other.vardas_;
00043         pavarde_ = other.pavarde_;
00044         nd_ = other.nd_;
00045         egzaminas_ = other.egzaminas_;
00046         galutinis_ = other.galutinis_;
00047         return *this;
00048     }
00049
00050     ~Studentas() {
00051         nd_.clear();
00052     }
00053
00054
00055
00056     const vector<int>& getNamuDarbai() const { return nd_; }
00057     int getEgzaminas() const { return egzaminas_; }
00058     double getGalutinis() const { return galutinis_; }
00059
00060     void setGalutinis(double g) { galutinis_ = g; }
00061     void setNamuDarbai(const vector<int>& nd) { nd_ = nd; }
00062     void setEgzaminas(int egzaminas) { egzaminas_ = egzaminas; }
00063
00064     void isvestiInformacija() const override {
00065         cout << "Studentas: " << vardas_ << " " << pavarde_
00066              << ", galutinis balas: " << galutinis_ << endl;
00067     }
00068
00069     static double galutinisVidurkis(const vector<int>& namu_darbai, int egzaminas);
00070     static double galutineMediana(vector<int> namu_darbai, int egzaminas);
00071
00072
00073
00074     friend ostream& operator<<(ostream& out, const Studentas& s) {
00075         if (&out == &cout) {
00076             out << "Vardas: " << s.vardas_ << ", Pavarde: " << s.pavarde_ << "\n";
00077             out << "Egzaminas: " << s.egzaminas_ << "\n";
00078             out << "Namu darbu pazymiai: ";
00079             for (int nd : s.nd_) out << nd << " ";
00080             out << "\nGalutinis: " << fixed << setprecision(2) << s.galutinis_ << "\n";
00081         }
00082         else {
00083             out << left << setw(15) << s.pavarde_
00084                  << setw(15) << s.vardas_;
00085             for (int nd : s.nd_) {
00086                 out << setw(5) << nd;
00087             }
00088             out << setw(10) << s.egzaminas_;
00089         }
00090         return out;
00091     }
00092
00093
00094
00095     friend istream& operator>>(istream& in, Studentas& s) {
00096         if (&in == &cin) {
00097             cout << "Iveskite studento pavarde: ";
00098         }
00099         in >> s.pavarde_;
00100
00101         if (&in == &cin) {
00102             cout << "Iveskite studento varda: ";
00103         }
00104         in >> s.vardas_;
00105
00106         if (&in == &cin) {
00107             cout << "Iveskite namu darbu balus (ivesti neigiama skaiciu, norint baigti):\n";
00108         }

```

```

00109         s.nd_.clear();
00110         int balas;
00111         while (in » balas && balas >= 0) {
00112             s.nd_.push_back(balas);
00113             if (&in == &cin) {
00114                 cout « "Namu darbu balas: ";
00115             }
00116         }
00117
00118         if (&in == &cin) {
00119             cout « "Iveskite egzamino bala: ";
00120         }
00121         in.clear();
00122         in » s.egzaminas_;
00123
00124         s.galutinis_ = Studentas::galutinisVidurkis(s.nd_, s.egzaminas_);
00125
00126         return in;
00127     }
00128
00129
00130
00131
00132     static Studentas generuotiStudentuDuomenis() {
00133         Studentas s;
00134         s.vardas_ = "Vardas" + to_string(rand() % 100 + 1);
00135         s.pavarde_ = "Pavarde" + to_string(rand() % 100 + 1);
00136         s.egzaminas_ = rand() % 10 + 1;
00137
00138         for (int i = 0; i < 3; i++) {
00139             s.nd_.push_back(rand() % 10 + 1);
00140         }
00141
00142         s.galutinis_ = Studentas::galutinisVidurkis(s.nd_, s.egzaminas_);
00143         return s;
00144     }
00145
00146
00147     static vector<Studentas> nuskaitytiIsFailo(const string& failoVardas) {
00148         ifstream failas(failoVardas);
00149         vector<Studentas> studentai;
00150
00151         if (!failas) {
00152             cerr « "Klaida: Nepavyko atidaryti failo '" « failoVardas « "'.\n";
00153             return studentai;
00154         }
00155
00156         string praleistiEilute;
00157         getline(failas, praleistiEilute);
00158         Studentas s;
00159         while (failas » s) {
00160             studentai.push_back(s);
00161         }
00162
00163         if (failas.bad()) {
00164             cerr « "Klaida: Failo skaitymo metu ávyko klaida.\n";
00165         }
00166
00167         failas.close();
00168         return studentai;
00169     }
00170
00171
00172     static void rasytiIFaila(const vector<Studentas>& studentai, const string& failoVardas) {
00173         ofstream failas(failoVardas);
00174         if (!failas) {
00175             cerr « "Nepavyko sukurti failo: " « failoVardas « endl;
00176             return;
00177         }
00178
00179         failas « left « setw(15) « "Pavarde"
00180             « setw(15) « "Vardas"
00181             « setw(5) « "ND1"
00182             « setw(5) « "ND2"
00183             « setw(5) « "ND3"
00184             « setw(10) « "Egzaminas" « endl;
00185
00186         for (const auto& s : studentai) {
00187             failas « s « endl;
00188         }
00189
00190         failas.close();
00191     }
00192 };
00193
00194 void isvestis(const vector<Studentas>& studentai, int pasirinkimas);
00195 void nuskaitytiIsFailo(const string& failoPavadinimas, vector<Studentas>& studentai);

```



```
00196 void generuotiAtsitiktiniusRezultatus(Studentas& s);
00197 bool palyginimas(const Studentas& a, const Studentas& b);
00198 void rusiavimas(vector<Studentas>& studentai);
00199
00200 #endif
```



# Index

- ~Studentas
  - Studentas, [8](#)
- ~Zmogus
  - Zmogus, [11](#)
- atmintis
  - failai.cpp, [13](#)
  - failai.h, [15](#)
- atmintisList
  - list.cpp, [18](#)
  - list.h, [20](#)
- failai.cpp, [13](#)
  - atmintis, [13](#)
  - generuotiStudentuDuomenis, [13](#)
  - matuotiVeikimoGreiti, [13](#)
  - nuskaitytiStudentus, [14](#)
  - palyginimas\_pavarde, [14](#)
  - palyginimas\_pazimys, [14](#)
  - palyginimas\_vardas, [14](#)
  - rusiavimas\_pavarde, [14](#)
  - rusiavimas\_pazimys, [14](#)
  - rusiavimas\_vardas, [14](#)
  - rusiuotiStudentus, [15](#)
  - surusioti\_failai, [15](#)
- failai.h, [15](#)
  - atmintis, [15](#)
  - generuotiStudentuDuomenis, [15](#)
  - matuotiVeikimoGreiti, [16](#)
  - palyginimas\_pavarde, [16](#)
  - palyginimas\_pazimys, [16](#)
  - palyginimas\_vardas, [16](#)
  - rusiavimas\_pavarde, [16](#)
  - rusiavimas\_pazimys, [16](#)
  - rusiavimas\_vardas, [16](#)
  - rusiuotiStudentus, [16](#)
  - surusioti\_failai, [17](#)
- galutineMediana
  - Studentas, [9](#)
- galutinisVidurkis
  - Studentas, [9](#)
- generuotiAtsitiktiniusRezultatus
  - stud.cpp, [23](#)
  - stud.h, [24](#)
- generuotiStudentuDuomenis
  - failai.cpp, [13](#)
  - failai.h, [15](#)
  - Studentas, [9](#)
- getEgzaminas
  - Studentas, [9](#)
- getGalutinis
  - Studentas, [9](#)
- getNamuDarbai
  - Studentas, [9](#)
- getPavarde
  - Zmogus, [11](#)
- getVardas
  - Zmogus, [11](#)
- idk.cpp, [17](#)
  - main, [18](#)
- isvestiInformacija
  - Studentas, [9](#)
  - Zmogus, [12](#)
- isvestis
  - stud.cpp, [23](#)
  - stud.h, [24](#)
- isvestisList
  - list.cpp, [18](#)
  - list.h, [20](#)
- list.cpp, [18](#)
  - atmintisList, [18](#)
  - isvestisList, [18](#)
  - matuotiVeikimoGreitiList, [18](#)
  - nuskaitytiIsFailoList, [18](#)
  - rusiavimas\_pavardeList, [19](#)
  - rusiavimas\_pazimysList, [19](#)
  - rusiavimas\_vardasList, [19](#)
  - rusiavimasList, [19](#)
  - rusiuotiStudentusList, [19](#)
  - surusioti\_failaiList, [19](#)
- list.h, [20](#)
  - atmintisList, [20](#)
  - isvestisList, [20](#)
  - matuotiVeikimoGreitiList, [20](#)
  - nuskaitytiIsFailoList, [20](#)
  - rusiavimas\_pavardeList, [20](#)
  - rusiavimas\_pazimysList, [21](#)
  - rusiavimas\_vardasList, [21](#)
  - rusiavimasList, [21](#)
  - rusiuotiStudentusList, [21](#)
  - surusioti\_failaiList, [21](#)
- main
  - idk.cpp, [18](#)
- matuotiVeikimoGreiti
  - failai.cpp, [13](#)
  - failai.h, [16](#)

- matuotiVeikimoGreitiList
  - list.cpp, [18](#)
  - list.h, [20](#)
- mylib.h, [22](#)
- nuskaitytilsFailo
  - stud.cpp, [23](#)
  - stud.h, [24](#)
  - Studentas, [9](#)
- nuskaitytilsFailoList
  - list.cpp, [18](#)
  - list.h, [20](#)
- nuskaitytiStudentus
  - failai.cpp, [14](#)
- operator<<
  - Studentas, [10](#)
- operator>>
  - Studentas, [10](#)
- operator=
  - Studentas, [9](#)
- palyginimas
  - stud.cpp, [23](#)
  - stud.h, [24](#)
- palyginimas\_pavarde
  - failai.cpp, [14](#)
  - failai.h, [16](#)
- palyginimas\_pazimys
  - failai.cpp, [14](#)
  - failai.h, [16](#)
- palyginimas\_vardas
  - failai.cpp, [14](#)
  - failai.h, [16](#)
- pavarde\_
  - Zmogus, [12](#)
- rasytilFaila
  - Studentas, [10](#)
- rusiavimas
  - stud.cpp, [23](#)
  - stud.h, [24](#)
- rusiavimas\_pavarde
  - failai.cpp, [14](#)
  - failai.h, [16](#)
- rusiavimas\_pavardeList
  - list.cpp, [19](#)
  - list.h, [20](#)
- rusiavimas\_pazimys
  - failai.cpp, [14](#)
  - failai.h, [16](#)
- rusiavimas\_pazimysList
  - list.cpp, [19](#)
  - list.h, [21](#)
- rusiavimas\_vardas
  - failai.cpp, [14](#)
  - failai.h, [16](#)
- rusiavimas\_vardasList
  - list.cpp, [19](#)
- list.h, [21](#)
- rusiavimasList
  - list.cpp, [19](#)
  - list.h, [21](#)
- rusiuotiStudentus
  - failai.cpp, [15](#)
  - failai.h, [16](#)
- rusiuotiStudentusList
  - list.cpp, [19](#)
  - list.h, [21](#)
- setEgzaminas
  - Studentas, [10](#)
- setGalutinis
  - Studentas, [10](#)
- setNamuDarbai
  - Studentas, [10](#)
- stud.cpp, [22](#)
  - generuotiAtsitiktiniusRezultatus, [23](#)
  - isvestis, [23](#)
  - nuskaitytilsFailo, [23](#)
  - palyginimas, [23](#)
  - rusiavimas, [23](#)
- stud.h, [23](#)
  - generuotiAtsitiktiniusRezultatus, [24](#)
  - isvestis, [24](#)
  - nuskaitytilsFailo, [24](#)
  - palyginimas, [24](#)
  - rusiavimas, [24](#)
- Studentas, [7](#)
  - ~Studentas, [8](#)
  - galutineMediana, [9](#)
  - galutinisVidurkis, [9](#)
  - generuotiStudentuDuomenis, [9](#)
  - getEgzaminas, [9](#)
  - getGalutinis, [9](#)
  - getNamuDarbai, [9](#)
  - isvestiInformacija, [9](#)
  - nuskaitytilsFailo, [9](#)
  - operator<<, [10](#)
  - operator>>, [10](#)
  - operator=, [9](#)
  - rasytilFaila, [10](#)
  - setEgzaminas, [10](#)
  - setGalutinis, [10](#)
  - setNamuDarbai, [10](#)
  - Studentas, [8](#)
- surusioti\_failai
  - failai.cpp, [15](#)
  - failai.h, [17](#)
- surusioti\_failaiList
  - list.cpp, [19](#)
  - list.h, [21](#)
- vardas\_
  - Zmogus, [12](#)
- Zmogus, [11](#)
  - ~Zmogus, [11](#)

getPavarde, [11](#)  
getVardas, [11](#)  
isvestiInformacija, [12](#)  
pavarde\_, [12](#)  
vardas\_, [12](#)  
Zmogus, [11](#)