Student

Generated by Doxygen 1.13.2

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 Human Class Reference . . . 7

4.1.1 Detailed Description . . . 7

4.1.2 Constructor & Destructor Documentation . . . 8

4.1.2.1 Human() [1/2] . . . 8

4.1.2.2 Human() [2/2] . . . 8

4.1.2.3 ∼Human() . . . 8

4.1.3 Member Function Documentation . . . 8

4.1.3.1 calculateFinalMean() . . . 8

4.1.3.2 calculateFinalMedian() . . . 8

4.1.3.3 getName() . . . 8

4.1.3.4 getSurname() . . . 8

4.1.3.5 setName() . . . 9

4.1.3.6 setSurname() . . . 9

4.1.4 Member Data Documentation . . . 9

4.1.4.1 name_ . . . 9

4.1.4.2 surname_ . . . 9

4.2 Student Class Reference . . . 9

4.2.1 Detailed Description . . . 10

4.2.2 Constructor & Destructor Documentation . . . 11

4.2.2.1 Student() [1/6] . . . 11

4.2.2.2 Student() [2/6] . . . 11

4.2.2.3 Student() [3/6] . . . 11

4.2.2.4 Student() [4/6] . . . 11

4.2.2.5 Student() [5/6] . . . 12

4.2.2.6 Student() [6/6] . . . 12

4.2.2.7 ∼Student() . . . 12

4.2.3 Member Function Documentation . . . 13

4.2.3.1 calculateFinalMean() . . . 13

4.2.3.2 calculateFinalMedian() . . . 13

4.2.3.3 getExamMark() . . . 13

4.2.3.4 getMarks() . . . 13

4.2.3.5 operator=() [1/2] . . . 13

4.2.3.6 operator=() [2/2] . . . 13

4.2.3.7 output() 14
4.2.3.8 setExamMark() 14
4.2.3.9 setMarks() 14
4.2.4 Friends And Related Symbol Documentation 14
4.2.4.1 operator<< 14
4.2.4.2 operator>> 14

5 File Documentation **17**
5.1 studentas.cpp File Reference 17
5.1.1 Function Documentation 17
5.1.1.1 operator<<() 17
5.1.1.2 operator>>() 17
5.2 studentas.cpp 18
5.3 studentas.h File Reference 20
5.3.1 Function Documentation 20
5.3.1.1 InputExamMark() 20
5.3.1.2 InputMark() 20
5.4 studentas.h 21

Index **23**

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

Human 7
    Student 9

Chapter 2

Class Index

2.1 Class List

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

Human 7
Student 9

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

studentas.cpp 17
studentas.h 20

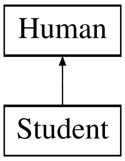
Chapter 4

Class Documentation

4.1 Human Class Reference

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

Inheritance diagram for Human:



Public Member Functions

- Human ()
- Human (wstring name, wstring surname)
- virtual ∼Human ()
- std::wstring getName () const
- std::wstring getSurname () const
- void setName (std::wstring name)
- void setSurname (std::wstring surname)
- virtual float calculateFinalMean () const =0
- virtual float calculateFinalMedian () const =0

Protected Attributes

- std::wstring name_
- std::wstring surname_

4.1.1 Detailed Description

Definition at line 21 of file studentas.h.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 Human() [1/2]

```
Human::Human ()  [inline]
```

Definition at line 26 of file studentas.h.

4.1.2.2 Human() [2/2]

```
Human::Human (
            wstring name,
            wstring surname)  [inline]
```

Definition at line 27 of file studentas.h.

4.1.2.3 ∼Human()

```
Human::∼Human ()  [virtual]
```

Definition at line 3 of file studentas.cpp.

4.1.3 Member Function Documentation

4.1.3.1 calculateFinalMean()

```
virtual float Human::calculateFinalMean () const  [pure virtual]
```

Implemented in Student.

4.1.3.2 calculateFinalMedian()

```
virtual float Human::calculateFinalMedian () const  [pure virtual]
```

Implemented in Student.

4.1.3.3 getName()

```
std::wstring Human::getName () const  [inline]
```

Definition at line 30 of file studentas.h.

4.1.3.4 getSurname()

```
std::wstring Human::getSurname () const  [inline]
```

Definition at line 31 of file studentas.h.

4.1.3.5 setName()

```
void Human::setName (
             std::wstring name)  [inline]
```

Definition at line 33 of file studentas.h.

4.1.3.6 setSurname()

```
void Human::setSurname (
             std::wstring surname)  [inline]
```

Definition at line 34 of file studentas.h.

4.1.4 Member Data Documentation

4.1.4.1 name_

```
std::wstring Human::name_  [protected]
```

Definition at line 23 of file studentas.h.

4.1.4.2 surname_

```
std::wstring Human::surname_  [protected]
```

Definition at line 24 of file studentas.h.

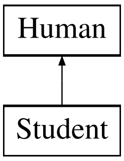
The documentation for this class was generated from the following files:

- studentas.h
- studentas.cpp

4.2 Student Class Reference

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

Inheritance diagram for Student:



Public Member Functions

- Student ()
- Student (int pazymiuSk)
- Student (wstring name, wstring surname, int pazymiuSk)
- Student (wstring name, wstring surname, vector< int > marks, int examMark)
- Student (const Student &orig)
- Student & operator= (const Student &orig)
- Student (Student &&orig)
- Student & operator= (Student &&orig)
- ∼Student ()
- std::vector< int > getMarks () const
- int getExamMark () const
- void setMarks (std::vector< int > marks)
- void setExamMark (int examMark)
- float calculateFinalMean () const
- float calculateFinalMedian () const
- wstring output ()

Public Member Functions inherited from Human

- Human ()
- Human (wstring name, wstring surname)
- virtual ∼Human ()
- std::wstring getName () const
- std::wstring getSurname () const
- void setName (std::wstring name)
- void setSurname (std::wstring surname)

Friends

- std::wostream & operator<< (std::wostream &out, const Student &st)
- std::wistream & operator>> (std::wistream &in, Student &st)

Additional Inherited Members

Protected Attributes inherited from Human

- std::wstring name_
- std::wstring surname_

4.2.1 Detailed Description

Definition at line 40 of file studentas.h.

4.2.2 Constructor & Destructor Documentation

4.2.2.1 Student() [1/6]

```
Student::Student ()  [inline]
```

Definition at line 45 of file studentas.h.

4.2.2.2 Student() [2/6]

```
Student::Student (
                 int pazymiuSk)
```

Constructor with 1 parameter

Parameters

pazymiuSk	- number of marks to generate randomly Both name and surname are generated randomly

Definition at line 45 of file studentas.cpp.

4.2.2.3 Student() [3/6]

```
Student::Student (
                 wstring name,
                 wstring surname,
                 int pazymiuSk)
```

Constructor with 3 parameters

Parameters

name	- name of the student
surname	- surname of the student
pazymiuSk	- number of marks to generate randomly

Definition at line 27 of file studentas.cpp.

4.2.2.4 Student() [4/6]

```
Student::Student (
                 wstring name,
                 wstring surname,
                 vector< int > marks,
                 int examMark)
```

Constructor with 4 parameters

Parameters

name	- name of the student
surname	- surname of the student
marks	- vector of marks
examMark	- exam mark

Definition at line 15 of file studentas.cpp.

4.2.2.5 Student() [5/6]

```
Student::Student (
         const Student & orig)
```

Copy constructor

Parameters

orig	- object to copy from

Definition at line 64 of file studentas.cpp.

4.2.2.6 Student() [6/6]

```
Student::Student (
         Student && orig)
```

Move constructor

Parameters

orig	- object to move from

Definition at line 87 of file studentas.cpp.

4.2.2.7 ∼Student()

```
Student::∼Student ()
```

Definition at line 174 of file studentas.cpp.

4.2.3 Member Function Documentation

4.2.3.1 calculateFinalMean()

```
float Student::calculateFinalMean () const  [virtual]
```

Function to calculate the final mean of the student
Returns the final mean of the student

Implements Human.

Definition at line 184 of file studentas.cpp.

4.2.3.2 calculateFinalMedian()

```
float Student::calculateFinalMedian () const  [virtual]
```

Function to calculate the final median of the student
Returns the final median of the student

Implements Human.

Definition at line 193 of file studentas.cpp.

4.2.3.3 getExamMark()

```
int Student::getExamMark () const  [inline]
```

Definition at line 61 of file studentas.h.

4.2.3.4 getMarks()

```
std::vector< int > Student::getMarks () const  [inline]
```

Definition at line 60 of file studentas.h.

4.2.3.5 operator=() [1/2]

```
Student & Student::operator= (
             const Student & orig)
```

Copy assignment operator

Parameters

orig	- object to copy from

Definition at line 74 of file studentas.cpp.

4.2.3.6 operator=() [2/2]

```
Student & Student::operator= (
             Student && orig)
```

Move assignment operator

Parameters

orig	- object to move from

Definition at line 101 of file studentas.cpp.

4.2.3.7 output()

```
wstring Student::output ()
```

Function to output the student object
Required for testing

Definition at line 206 of file studentas.cpp.

4.2.3.8 setExamMark()

```
void Student::setExamMark (
             int examMark)  [inline]
```

Definition at line 64 of file studentas.h.

4.2.3.9 setMarks()

```
void Student::setMarks (
             std::vector< int > marks)  [inline]
```

Definition at line 63 of file studentas.h.

4.2.4 Friends And Related Symbol Documentation

4.2.4.1 operator<<

```
std::wostream & operator<< (
             std::wostream & out,
             const Student & st)  [friend]
```

Output operator

Parameters

output	- output stream
st	- object to output Outputs the surname, name, final mean and final median of the student

Definition at line 119 of file studentas.cpp.

4.2.4.2 operator>>

```
std::wistream & operator>> (
             std::wistream & in,
             Student & st)  [friend]
```

Input operator

Parameters

in	- input stream
st	- object to input Inputs the name, surname, marks and exam mark of the student (only works for stdin)

Definition at line 131 of file studentas.cpp.

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

- studentas.h
- studentas.cpp

Chapter 5

File Documentation

5.1 studentas.cpp File Reference

```
#include "../include/studentas.h"
```

Functions

- std::wostream & operator<< (std::wostream &output, const Student &st)
- std::wistream & operator>> (std::wistream &in, Student &st)

5.1.1 Function Documentation

5.1.1.1 operator<<()

```
std::wostream & operator<< (
            std::wostream & output,
            const Student & st)
```

Output operator

Parameters

output	- output stream
st	- object to output Outputs the surname, name, final mean and final median of the student

Definition at line 119 of file studentas.cpp.

5.1.1.2 operator>>()

```
std::wistream & operator>> (
            std::wistream & in,
            Student & st)
```

Input operator

Parameters

in	- input stream
st	- object to input Inputs the name, surname, marks and exam mark of the student (only works for stdin)

Definition at line 131 of file studentas.cpp.

5.2 studentas.cpp

Go to the documentation of this file.

```
#include "../include/studentas.h"

Human::~Human() {
    name_ = L"";
    surname_ = L"";
}

Student::Student(wstring name, wstring surname, vector<int> marks, int examMark) {
    name_ = name;
    surname_ = surname;
    marks_ = marks;
    examMark_ = examMark;
}
Student::Student(wstring name, wstring surname, int pazymiuSk) {
    std::random_device rd;
    std::mt19937 mt(rd());
    std::uniform_int_distribution<int> dist(1, 10);
    for (int i = 0; i < pazymiuSk; i++) {
        marks_.push_back(dist(mt));
        std::wcout « L"Generuotas pazymys: " « marks_[i] « std::endl;
    }
    examMark_ = dist(mt);
    std::wcout « L"Generuotas egzamino pazymys: " « examMark_ « std::endl;
    name_ = name;
    surname_ = surname;
}
Student::Student(int pazymiuSk) {
    std::random_device rd;
    std::mt19937 mt(rd());
    std::uniform_int_distribution<int> dist(1, 10);
    for (int i = 0; i < pazymiuSk; i++) {
        marks_.push_back(dist(mt));
        std::wcout « L"Generuotas pazymys: " « marks_[i] « std::endl;
    }
    examMark_ = dist(mt);
    std::wcout « L"Generuotas egzamino pazymys: " « examMark_ « std::endl;
    name_ = L"name" + std::to_wstring(dist(mt));
    wcout « L"Generuotas vardas: " « name_ « endl;
    surname_ = L"surname" + std::to_wstring(dist(mt));
    wcout « L"Generuota pavarde: " « surname_ « endl;
}
Student::Student(const Student& orig) { //copy constructor
    name_ = orig.name_;
    surname_ = orig.surname_;
    marks_ = orig.marks_;
    examMark_ = orig.examMark_;
}
Student& Student::operator=(const Student& orig) { //copy assignment operator
    if (this != &orig) {
        name_ = orig.name_;
        surname_ = orig.surname_;
        marks_ = orig.marks_;
        examMark_ = orig.examMark_;
    }
    return *this;
}
Student::Student(Student&& orig) { //move constructor
    name_ = orig.name_;
    surname_ = orig.surname_;
    marks_ = std::move(orig.marks_);
    examMark_ = orig.examMark_;
    orig.name_ = L"";
    orig.surname_ = L"";
    orig.marks_.clear();
    orig.examMark_ = 0;
}
Student& Student::operator=(Student&& orig) { //move assignment operator
    if (&orig == this) return *this;
    name_ = orig.name_;
    surname_ = orig.surname_;
    marks_ = std::move(orig.marks_);
    examMark_ = orig.examMark_;
    orig.name_ = L"";
    orig.surname_ = L"";
    orig.marks_.clear();
    orig.examMark_ = 0;
    return *this;
}
std::wostream& operator«(std::wostream& output, const Student& st) {
    output « setw(17) « left « st.getSurname() « setw(17) « left
        « st.getName() « setw(20) « left « setprecision(2) « fixed
        « st.calculateFinalMean() « setw(15) « left « setprecision(2) « fixed « st.calculateFinalMedian() « L"\n";
    return output;
}
std::wistream& operator»(std::wistream& in, Student& st) {
    wcout « L"Iveskite vardą, pavardę\n";
    in » st.name_ » st.surname_;
    wcout « L"Iveskite pažymius (0 - pabaigti)\n";
    while (true) {
        try {
            int mark = InputMark();
            if (mark == 0) {
                if (st.marks_.size() != 0) {
                    break;
                }
                else throw L"\007Įveskite bent vieną pažymį";
                continue;
            }
            st.marks_.push_back(mark);
        }
        catch (const wchar_t* e) {
            wcerr « e « endl;
            continue;
        }
        catch (...) {
            wcerr « L"\007Nežinoma klaida" « endl;
            continue;
        }
    }
    wcout « L"Įveskite egzamino pažymį:\n";
    while (true) {
        try {
            st.examMark_ = InputExamMark();
            break;
        }
        catch (const wchar_t* e) {
            wcerr « e « endl;
            continue;
        }
        catch (...) {
            wcerr « L"\007Nežinoma klaida" « endl;
            continue;
        }
    }
    return in;
}

Student::~Student() {
    name_ = L"";
    surname_ = L"";
    marks_.clear();
    examMark_ = 0;
}
float Student::calculateFinalMean() const {
    vector<int> marks = getMarks();
    if (marks.size() == 0) return 0;
    return 0.4 * ((std::accumulate(marks.begin(), marks.end(), 0) * 1.0) / (marks.size() * 1.0)) + 0.6 * examMark_;
}
float Student::calculateFinalMedian() const {
    vector<int> marks = getMarks();
    if (marks.size() == 0) return 0;
    float median = 0;
    std::sort(marks.begin(), marks.end());
    (marks.size() % 2 != 0) ? median = marks[marks.size() / 2] : median = (marks[marks.size() / 2] + marks[marks.size() / 2 - 1]) / 2.0;
    median = median * 0.4 + examMark_ * 0.6;
    return median;
}
```

```
wstring Student::output() {
    wstring output = name_ + surname_;
    for (int i = 0; i < marks_.size(); i++) {
        output += std::to_wstring(marks_[i]) + L" ";
    }
    output += std::to_wstring(examMark_);
    return output;
}
```

5.3 studentas.h File Reference

```
#include <string>
#include <vector>
#include <numeric>
#include <algorithm>
#include <random>
#include <iostream>
#include <iomanip>
```

Classes

- class Human
- class Student

Functions

- int InputMark ()
- int InputExamMark ()

5.3.1 Function Documentation

5.3.1.1 InputExamMark()

```
int InputExamMark ()
```

5.3.1.2 InputMark()

```
int InputMark ()
```

5.4 studentas.h

Go to the documentation of this file.

```
#pragma once
#include <string>
#include <vector>
#include <numeric>
#include <algorithm>
#include <random>
#include <iostream>
#include <iomanip>
#include <algorithm>

using std::vector;
using std::wcout;
using std::endl;
using std::wstring;
using std::left;
using std::setprecision;
using std::setw;
using std::fixed;
using std::wcerr;

class Human {
protected:
    std::wstring name_;
    std::wstring surname_;
public:
    Human() : name_(L""), surname_(L"") {}
    Human(wstring name, wstring surname) : name_(name), surname_(surname) {}
    virtual ~Human();

    inline std::wstring getName() const { return name_; }
    inline std::wstring getSurname() const { return surname_; }

    void setName(std::wstring name) { name_ = name; }
    void setSurname(std::wstring surname) { surname_ = surname; }

    virtual float calculateFinalMean() const = 0;
    virtual float calculateFinalMedian() const = 0;
};

class Student : public Human {
private:
    std::vector<int>marks_;
    int examMark_;
public:
    Student() : Human(), marks_(), examMark_(1) {}
    Student(int pazymiuSk);
    Student(wstring name, wstring surname, int pazymiuSk);
    Student(wstring name, wstring surname, vector<int> marks, int examMark);

    Student(const Student& orig); //copy constructor
    Student& operator=(const Student& orig); //copy assignment operator
    Student(Student&& orig);
    Student& operator=(Student&& orig);

    friend std::wostream& operator«(std::wostream& out, const Student& st);
    friend std::wistream& operator»(std::wistream& in, Student& st);

    ~Student();

    inline std::vector<int> getMarks() const { return marks_; }
    inline int getExamMark() const { return examMark_; }

    void setMarks(std::vector<int> marks) { marks_ = marks; }
    void setExamMark(int examMark) { examMark_ = examMark; }

    float calculateFinalMean() const;
    float calculateFinalMedian() const;

    wstring output();
};
int InputMark();
int InputExamMark();
```

Index

∼Human
  Human, 8
∼Student
  Student, 12

calculateFinalMean
  Human, 8
  Student, 13
calculateFinalMedian
  Human, 8
  Student, 13

getExamMark
  Student, 13
getMarks
  Student, 13
getName
  Human, 8
getSurname
  Human, 8

Human, 7
  ∼Human, 8
  calculateFinalMean, 8
  calculateFinalMedian, 8
  getName, 8
  getSurname, 8
  Human, 8
  name_, 9
  setName, 8
  setSurname, 9
  surname_, 9

InputExamMark
  studentas.h, 20
InputMark
  studentas.h, 20

name_
  Human, 9

operator<<
  Student, 14
  studentas.cpp, 17
operator>>
  Student, 14
  studentas.cpp, 17
operator=
  Student, 13
output
  Student, 14

setExamMark
  Student, 14
setMarks
  Student, 14
setName
  Human, 8
setSurname
  Human, 9
Student, 9
  ∼Student, 12
  calculateFinalMean, 13
  calculateFinalMedian, 13
  getExamMark, 13
  getMarks, 13
  operator<<, 14
  operator>>, 14
  operator=, 13
  output, 14
  setExamMark, 14
  setMarks, 14
  Student, 11, 12
studentas.cpp, 17
  operator<<, 17
  operator>>, 17
studentas.h, 20
  InputExamMark, 20
  InputMark, 20
surname_
  Human, 9