Student Managment System

Generated by Doxygen 1.12.0

Chapter 1

Class Documentation

1.1 ENSTA_Student Struct Reference

Structure representing a student.

Public Attributes

- int id
- char name [MAX_NAME_LENGTH]
- int birthYear
- char class [3]
- float grades [NUM_MODULES]
- float average
- bool deleted

1.1.1 Detailed Description

Structure representing a student.

1.1.2 Member Data Documentation

1.1.2.1 average

```
float ENSTA_Student::average
```

1.1.2.2 birthYear

```
int ENSTA_Student::birthYear
```

1.1.2.3 class

```
char ENSTA_Student::class[3]
```

1.1.2.4 deleted

```
bool ENSTA_Student::deleted
```

1.1.2.5 grades

```
float ENSTA_Student::grades[NUM_MODULES]
```

1.1.2.6 id

```
int ENSTA_Student::id
```

1.1.2.7 name

```
char ENSTA_Student::name[MAX_NAME_LENGTH]
```

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

- functions.c

Chapter 2

File Documentation

2.1 functions.c File Reference

This is the main file for the ENSTA Student Management System. It contains the main menu and the implementation of the functions to manage students.

```
#include <stdio.h>
#include <string.h>
#include <stdbool.h>
#include <ctype.h>
#include "functions.h"
```

Classes

- struct ENSTA_Student

  Structure representing a student.

Macros

- #define FILE_NAME "ENSTA_Students.txt"
- #define NUM_MODULES 4
- #define MAX_NAME_LENGTH 40

Functions

- void add_student ()

  Procedure to add a student to the file.
- ENSTA_Student * student_exists (int id)

  Function to check if a student with a given ID exists.
- void search_student ()

  Procedure to search for a student by ID.
- void modify_student ()

  Procedure to modify a student's information.
- void delete_student ()

Procedure to delete a student from the file.

- void uppercase (char class[3])

  Additional procedure sed in extract_by_class() to uppercase the class name (could be unnecessary?)
- void extract_by_class ()

  Procedure to extract students by class.
- void reorganize_file ()

  Procedure to reorganize the file by removing logically deleted entries.
- float grade_average (float grades[4])

  Function to calculate the average of the grades.
- int main ()
- float grade_average (float grades[NUM_MODULES])

Variables

- const char * MODULE_NAMES [ ] = {"SFSD", "OOP", "Mathematical Analysis", "Linear Algebra"}
- const int MODULE_COEFFICIENTS [ ] = {4, 3, 5, 2}

2.1.1 Detailed Description

This is the main file for the ENSTA Student Management System. It contains the main menu and the implementation of the functions to manage students.

Authors

BOUSSEKINE Mohamed Ismail, NACERI Rim Serine, FERKIOUI Akram, HARIZI Raouank, AMEDHKOUH Darine, HAMMOUTI Walid

Date

2024-12-27

2.1.2 Macro Definition Documentation

2.1.2.1 FILE_NAME

```
#define FILE_NAME "ENSTA_Students.txt"
```

2.1.2.2 MAX_NAME_LENGTH

```
#define MAX_NAME_LENGTH 40
```

2.1.2.3 NUM_MODULES

```
#define NUM_MODULES 4
```

2.1.3 Function Documentation

2.1.3.1 add_student()

```
void add_student ()
```

Procedure to add a student to the file.

Author

Akram

2.1.3.2 delete_student()

```
void delete_student ()
```

Procedure to delete a student from the file.

Author

Raounek

2.1.3.3 extract_by_class()

```
void extract_by_class ()
```

Procedure to extract students by class.

Author

Serine

2.1.3.4 grade_average() [1/2]

```
float grade_average (
            float grades[4])
```

Function to calculate the average of the grades.

Author

Darine

Parameters

grades	The grades of the student

Returns

The average of the grades

2.1.3.5 grade_average() [2/2]

```
float grade_average (
            float grades[NUM_MODULES])
```

2.1.3.6 main()

```
int main ()
```

2.1.3.7 modify_student()

```
void modify_student ()
```

Procedure to modify a student's information.

Author

Soundouss

2.1.3.8 reorganize_file()

```
void reorganize_file ()
```

Procedure to reorganize the file by removing logically deleted entries.

Author

Walid

2.1.3.9 search_student()

```
void search_student ()
```

Procedure to search for a student by ID.

Author

Ismail

2.1.3.10 student_exists()

```
ENSTA_Student * student_exists (
            int id)
```

Function to check if a student with a given ID exists.

Author

Ismail

Parameters

id	The ID of the student to check

Returns

A pointer to the student if found, NULL otherwise

2.1.3.11 uppercase()

```
void uppercase (
            char class[3])
```

Additional procedure sed in extract_by_class() to uppercase the class name (could be unnecessary?)

Author

Serine

Parameters

class	The class name to extract students from

2.1.4 Variable Documentation

2.1.4.1 MODULE_COEFFICIENTS

```
const int MODULE_COEFFICIENTS[] = {4, 3, 5, 2}
```

2.1.4.2 MODULE_NAMES

```
const char* MODULE_NAMES[] = {"SFSD", "OOP", "Mathematical Analysis", "Linear Algebra"}
```