

IZP - Projekt 2

V1

Generated by Doxygen 1.8.20

1 Todo List	1
2 Data Structure Index	3
2.1 Data Structures	3
3 File Index	5
3.1 File List	5
4 Data Structure Documentation	7
4.1 Cell Struct Reference	7
4.1.1 Detailed Description	7
4.1.2 Field Documentation	7
4.1.2.1 content	7
4.2 Row Struct Reference	8
4.2.1 Detailed Description	8
4.2.2 Field Documentation	8
4.2.2.1 cells	8
4.2.2.2 num_of_cells	8
4.3 Table Struct Reference	9
4.3.1 Detailed Description	9
4.3.2 Field Documentation	9
4.3.2.1 num_of_rows	9
4.3.2.2 rows	9
5 File Documentation	11
5.1 CMakeLists.txt File Reference	11
5.2 sps.c File Reference	11
5.2.1 Macro Definition Documentation	12
5.2.1.1 BLACKLISTED_DELIMS	12
5.2.2 Enumeration Type Documentation	12
5.2.2.1 ErrorCodes	12
5.2.3 Function Documentation	12
5.2.3.1 check_sanity_of_delims()	12
5.2.3.2 load_delims_from_args()	13
5.2.3.3 load_table()	14
5.2.3.4 main()	15
5.2.3.5 strings_equal()	15
Index	17

Chapter 1

Todo List

Global `main` (int argc, char *argv[])

Refactor error handling

Chapter 2

Data Structure Index

2.1 Data Structures

Here are the data structures with brief descriptions:

Cell	Store for data of single cell	7
Row	Store for data of single row	8
Table	Store for data of whole table	9

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

sps.c	11
---------------------------------	----

Chapter 4

Data Structure Documentation

4.1 Cell Struct Reference

Store for data of single cell.

Data Fields

- char * [content](#)

4.1.1 Detailed Description

Store for data of single cell.

4.1.2 Field Documentation

4.1.2.1 content

```
char* content
```

Raw content of single cell

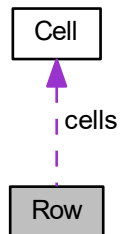
The documentation for this struct was generated from the following file:

- [sps.c](#)

4.2 Row Struct Reference

Store for data of single row.

Collaboration diagram for Row:



Data Fields

- int [num_of_cells](#)
- [Cell](#) * [cells](#)

4.2.1 Detailed Description

Store for data of single row.

4.2.2 Field Documentation

4.2.2.1 cells

```
Cell* cells
```

Pointer to first cell in row

4.2.2.2 num_of_cells

```
int num_of_cells
```

Number of cells in row

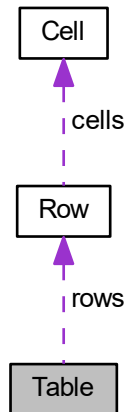
The documentation for this struct was generated from the following file:

- [sps.c](#)

4.3 Table Struct Reference

Store for data of whole table.

Collaboration diagram for Table:



Data Fields

- int [num_of_rows](#)
- [Row](#) * [rows](#)

4.3.1 Detailed Description

Store for data of whole table.

4.3.2 Field Documentation

4.3.2.1 num_of_rows

```
int num_of_rows
```

Number of rows in table

4.3.2.2 rows

```
Row* rows
```

Pointer to first row in table

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

- [sps.c](#)

Chapter 5

File Documentation

5.1 CMakeLists.txt File Reference

5.2 sps.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "stdbool.h"
```

Data Structures

- struct [Cell](#)
Store for data of single cell.
- struct [Row](#)
Store for data of single row.
- struct [Table](#)
Store for data of whole table.

Macros

- #define [BLACKLISTED_DELIMS](#) "\\\""

Enumerations

- enum [ErrorCodes](#) {
 [NO_ERROR](#), [MISSING_ARGS](#), [INVALID_DELIMITER](#), [TABLE_LOAD_ERROR](#),
 [FILE_DOESNT_EXIST](#) }
Flags that will be returned on error.

Functions

- `_Bool strings_equal` (const char *s1, const char *s2)
- int `load_table` (char *delims, char *filepath, [Table](#) *table)
- `_Bool check_sanity_of_delims` (char *delims)
- `_Bool load_delims_from_args` (char *argv[], char **delims)
- int `main` (int argc, char *argv[])

5.2.1 Macro Definition Documentation

5.2.1.1 BLACKLISTED_DELIMS

```
#define BLACKLISTED_DELIMS "\"\\\""
```

Character that are not allowed to use as delim character

5.2.2 Enumeration Type Documentation

5.2.2.1 ErrorCodes

```
enum ErrorCodes
```

Flags that will be returned on error.

Enumerator

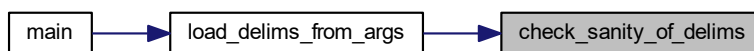
NO_ERROR	No error detected (default return code)
MISSING_ARGS	Some arguments are missing
INVALID_DELIMITER	Found invalid character in delimiters from arguments
TABLE_LOAD_ERROR	Data load error
FILE_DOESNT_EXIST	Error if input file doesnt exist or is used by another process

5.2.3 Function Documentation

5.2.3.1 check_sanity_of_delims()

```
_Bool check_sanity_of_delims (
    char * delims )
```


Here is the caller graph for this function:



5.2.3.2 load_delims_from_args()

```
_Bool load_delims_from_args (  
    char * argv[],  
    char ** delims )
```

Load delims

Check if second argument is -d flag and then set next argument as delims string

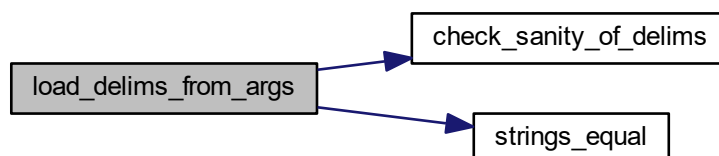
Parameters

<i>argv</i>	Array of arguments
<i>delims</i>	pointer to array of delims

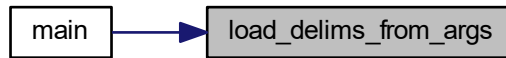
Returns

true if flag not found or delims after flag are valid, false if there are invalid characters in delims

Here is the call graph for this function:



Here is the caller graph for this function:



5.2.3.3 load_table()

```
int load_table (
    char * delims,
    char * filepath,
    Table * table )
```

Load table from file

Load and parse data from file

Parameters

<i>delims</i>	Array with all possible delimiters
<i>filepath</i>	Path to input file
<i>table</i>	Pointer table where data will be saved

Returns

NO_ERROR if table data are loaded and parsed properly, in other scenarios return error code

Here is the caller graph for this function:



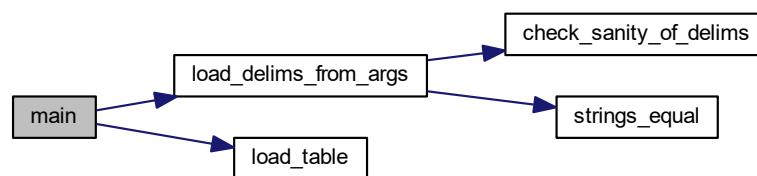
5.2.3.4 main()

```
int main (  
    int argc,  
    char * argv[] )
```

Main of whole program

Todo Refactor error handling

Here is the call graph for this function:



5.2.3.5 strings_equal()

```
_Bool strings_equal (  
    const char * s1,  
    const char * s2 )
```

Check if two string `s1` and `s2` are same

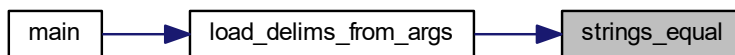
Parameters

<code>s1</code>	First string
<code>s2</code>	Second string

Returns

true if string are equal, false if dont

Here is the caller graph for this function:



Index

BLACKLISTED_DELIMS

sps.c, [12](#)

Cell, [7](#)

content, [7](#)

cells

Row, [8](#)

check_sanity_of_delims

sps.c, [12](#)

CMakeLists.txt, [11](#)

content

Cell, [7](#)

ErrorCodes

sps.c, [12](#)

FILE_DOESNT_EXIST

sps.c, [12](#)

INVALID_DELIMITER

sps.c, [12](#)

load_delims_from_args

sps.c, [13](#)

load_table

sps.c, [14](#)

main

sps.c, [14](#)

MISSING_ARGS

sps.c, [12](#)

NO_ERROR

sps.c, [12](#)

num_of_cells

Row, [8](#)

num_of_rows

Table, [9](#)

Row, [8](#)

cells, [8](#)

num_of_cells, [8](#)

rows

Table, [9](#)

sps.c, [11](#)

BLACKLISTED_DELIMS, [12](#)

check_sanity_of_delims, [12](#)

ErrorCodes, [12](#)

FILE_DOESNT_EXIST, [12](#)

INVALID_DELIMITER, [12](#)

load_delims_from_args, [13](#)

load_table, [14](#)

main, [14](#)

MISSING_ARGS, [12](#)

NO_ERROR, [12](#)

strings_equal, [15](#)

TABLE_LOAD_ERROR, [12](#)

strings_equal

sps.c, [15](#)

Table, [9](#)

num_of_rows, [9](#)

rows, [9](#)

TABLE_LOAD_ERROR

sps.c, [12](#)