

Repjegy Nagyhazi

1

Készítette Doxygen 1.8.14

Tartalomjegyzék

1. README	1
2. Adatszerkezet-mutató	3
2.1. Adatszerkezetek	3
3. Fájlmutató	5
3.1. Fájllista	5
4. Adatszerkezetek dokumentációja	7
4.1. FlightBook struktúrareferencia	7
4.1.1. Részletes leírás	7
4.2. FlightData struktúrareferencia	7
4.2.1. Részletes leírás	8
4.3. GenericItem struktúrareferencia	8
4.3.1. Részletes leírás	8
4.3.2. Adatmezők dokumentációja	8
4.3.2.1. next_item	8
4.3.2.2. value	8
4.4. GenericList struktúrareferencia	8
4.4.1. Részletes leírás	9
4.4.2. Adatmezők dokumentációja	9
4.4.2.1. append	9
4.4.2.2. get_item	9
4.4.2.3. insert	10
4.4.2.4. remove	10
4.4.2.5. size	10
4.5. OptionNode struktúrareferencia	11
4.5.1. Részletes leírás	11
4.5.2. Adatmezők dokumentációja	11
4.5.2.1. nextNode	11
4.5.2.2. value	11
4.6. RuntimeState struktúrareferencia	12
4.6.1. Részletes leírás	12

5. Fájlok dokumentációja	13
5.1. FlightBook.h fájlreferencia	13
5.1.1. Részletes leírás	13
5.1.2. Függvények dokumentációja	13
5.1.2.1. book_flight()	13
5.1.2.2. do_book_flight()	14
5.1.2.3. fb_alloc_book()	14
5.1.2.4. fb_free_book()	14
5.1.2.5. fb_load_from_file()	15
5.1.2.6. fb_save_to_file()	15
5.1.2.7. finalize_book()	15
5.2. FlightData.h fájlreferencia	15
5.2.1. Részletes leírás	16
5.2.2. Függvények dokumentációja	16
5.2.2.1. fd_add_flight()	16
5.2.2.2. fd_alloc_flight()	16
5.2.2.3. fd_load_from_file()	16
5.2.2.4. fd_save_to_file()	17
5.2.2.5. find_flight()	17
5.2.2.6. find_flight_by_flightnum()	17
5.2.2.7. list_flights()	18
5.2.2.8. new_fligth()	18
5.2.2.9. remove_flight()	18
5.3. GenericList.h fájlreferencia	18
5.3.1. Részletes leírás	19
5.3.2. Típusdefiníciók dokumentációja	19
5.3.2.1. GenericItem	19
5.3.2.2. GenericList	19
5.3.3. Függvények dokumentációja	19
5.3.3.1. gl_create_list()	20

5.3.3.2. <code>gl_delete_list()</code>	20
5.3.3.3. <code>gl_filter_text()</code>	20
5.4. <code>intelligent_menu.h</code> fájlreferencia	20
5.4.1. Részletes leírás	21
5.4.2. Függvények dokumentációja	21
5.4.2.1. <code>varg_input_menu()</code>	21
5.4.2.2. <code>varg_options_menu()</code>	22
5.4.2.3. <code>varg_options_menu_generator()</code>	22
5.4.2.4. <code>varg_options_yesno()</code>	23
5.5. <code>main.c</code> fájlreferencia	23
5.5.1. Részletes leírás	23
5.5.2. Függvények dokumentációja	24
5.5.2.1. <code>main()</code>	24
5.5.2.2. <code>win_init()</code>	24
5.6. <code>menu.h</code> fájlreferencia	24
5.6.1. Részletes leírás	25
5.6.2. Típusdefiníciók dokumentációja	25
5.6.2.1. <code>OptionNode</code>	25
5.6.3. Függvények dokumentációja	25
5.6.3.1. <code>menu_input()</code>	25
5.6.3.2. <code>menu_input_list()</code>	25
5.6.3.3. <code>menu_message()</code>	27
5.6.3.4. <code>menu_options()</code>	27
5.6.3.5. <code>menu_options_list()</code>	28
5.7. <code>string-utils.h</code> fájlreferencia	28
5.7.1. Részletes leírás	28
5.7.2. Függvények dokumentációja	28
5.7.2.1. <code>allocate_string_array()</code>	28
5.7.2.2. <code>free_string_array()</code>	29
5.7.2.3. <code>stra_max()</code>	29

5.7.2.4.	<code>strcnt()</code>	30
5.7.2.5.	<code>streqq()</code>	30
5.7.2.6.	<code>strlow()</code>	30
5.7.2.7.	<code>strrmc()</code>	31
5.7.2.8.	<code>strtrm()</code>	31
5.8.	Structs.h fájlreferencia	31
5.8.1.	Részletes leírás	32
5.8.2.	Típusdefiníciók dokumentációja	32
5.8.2.1.	FlightBook	32
5.8.2.2.	FlightData	32
5.8.2.3.	Menus	32
5.8.2.4.	RuntimeState	32
5.8.3.	Enumerációk dokumentációja	32
5.8.3.1.	Menus	32
Tárgymutató		33

1. fejezet

README

Repjegyek Házi feladat Prog. 1 tárgyra

Készítette Magyar Máté (Barrow099)

A program célja egy egyszerű repülőjegy kezelő rendszer megvalósítása C nyelven, parancssoros grafikus felülettel.

A program linux operációs rendszeren lett készítve, tesztelve, más platformon előfordulhatnak hibák.

Fordítás

A program a Czirkos Zoltán által készített C-Econio könyvtárat használja, a fordításhoz más nem szükséges.

Dokumentáció

A program dokumentációja a mellékelt refdoc.pdf-ben található, Doxygennel készült.

Kérdések, hibák, észrevételek

mate@barrow099.hu

2. fejezet

Adatszerkezet-mutató

2.1. Adatszerkezetek

Az összes adatszerkezet listája rövid leírásokkal:

FlightBook	7
FlightData	7
GenericItem	8
GenericList	8
OptionNode	11
RuntimeState	12

3. fejezet

Fájlmutató

3.1. Fájllista

Az összes dokumentált fájl listája rövid leírásokkal:

FlightBook.h	13
FlightData.h	15
GenericList.h	18
intelligent_menu.h	20
main.c	23
menu.h	24
string-utils.h	28
Structs.h	31

4. fejezet

Adatszerkezetek dokumentációja

4.1. FlightBook struktúrareferencia

```
#include <Structs.h>
```

Adatmezők

- string **name**
- string **email**
- string **flightnum**
- [Menus](#) menu
- int **seat**

4.1.1. Részletes leírás

Egy foglalás adatait tárol

Ez a dokumentáció a struktúráról a következő fájl alapján készült:

- [Structs.h](#)

4.2. FlightData struktúrareferencia

```
#include <Structs.h>
```

Adatmezők

- string **from**
- string **to**
- string **flighth_number**
- string **date**
- string **time**
- int **seats**
- int **seats_taken**

4.2.1. Részletes leírás

Egy járat adatait tárolja

Ez a dokumentáció a struktúráról a következő fájl alapján készült:

- [Structs.h](#)

4.3. GenericItem struktúrareferencia

```
#include <GenericList.h>
```

Adatmezők

- void * [value](#)
- struct [GenericItem](#) * [next_item](#)

4.3.1. Részletes leírás

Egy listaelem tárolására szolgáló struktúra Tartalmaz egy értékpointert és egy pointert a következő elemre

4.3.2. Adatmezők dokumentációja

4.3.2.1. next_item

```
struct GenericItem* GenericItem::next_item
```

Pointer a lista következő elemére

4.3.2.2. value

```
void* GenericItem::value
```

Értékpointer

Ez a dokumentáció a struktúráról a következő fájl alapján készült:

- [GenericList.h](#)

4.4. GenericList struktúrareferencia

```
#include <GenericList.h>
```

Adatmezők

- `GenericItem * HEAD`
- `void (*)(get_item)(struct GenericList *lst, int index)`
- `bool (*)(insert)(struct GenericList *list, void *item, int index)`
- `int (*)(size)(struct GenericList *list)`
- `bool (*)(append)(struct GenericList *list, void *item)`
- `void (*)(remove)(struct GenericList *list, int index)`

4.4.1. Részletes leírás

Egy generikus lista tárolására szolgáló struktúra. Tartalmaz egy pointert az első elemre és függvénypointereket a lista kezelésére az áttekinthetőség kedvéért.

4.4.2. Adatmezők dokumentációja

4.4.2.1. append

```
bool (* GenericList::append) (struct GenericList *list, void *item)
```

Elem hozzáfűzése a lista végéhez

Paraméterek

<i>list</i>	Lista pointer
<i>item</i>	Értékpontner

Visszatérési érték

true, ha sikerült a hozzáfűzés

4.4.2.2. get_item

```
void* (*)(GenericList::get_item) (struct GenericList *lst, int index)
```

Egy listaelem lekérése

Paraméterek

<i>lst</i>	A pointer a listára
<i>index</i>	Az elem indexe

Visszatérési érték

Egy void* az értékre vagy NULL

4.4.2.3. insert

```
bool(* GenericList::insert) (struct GenericList *list, void *item, int index)
```

Érték beszúrása a listába

Paraméterek

<i>list</i>	A pointer a listára
<i>item</i>	Értékpointer
<i>index</i>	A beszúrás helyének indexe

Visszatérési érték

true, ha sikeres volt a beszúrás

4.4.2.4. remove

```
void(* GenericList::remove) (struct GenericList *list, int index)
```

Egy eghatározott listaelem törlése

Paraméterek

<i>list</i>	A lista
<i>index</i>	A törölni kívánt elem indexe

4.4.2.5. size

```
int(* GenericList::size) (struct GenericList *list)
```

A lista méretének lekérése

Paraméterek

<i>list</i>	A lista
-------------	---------

Visszatérési érték

A lista mérete

Ez a dokumentáció a struktúráról a következő fájl alapján készült:

- [GenericList.h](#)

4.5. OptionNode struktúráreferencia

```
#include <menu.h>
```

Adatmezők

- struct [OptionNode](#) * [nextNode](#)
- string [value](#)

4.5.1. Részletes leírás

Linkelt lista a listás opciómegjelenítéshez

Figyelmeztetés

Belső használatra

4.5.2. Adatmezők dokumentációja

4.5.2.1. nextNode

```
struct OptionNode* OptionNode::nextNode
```

A következő elem

4.5.2.2. value

```
string OptionNode::value
```

sztring érték

Ez a dokumentáció a struktúráról a következő fájl alapján készült:

- [menu.h](#)

4.6. RuntimeState struktúrareferencia

```
#include <Structs.h>
```

Adatmezők

- [GenericList](#) * **flight_list**
- [GenericList](#) * **book_list**

4.6.1. Részletes leírás

A program futás közbeni állapotát tárolja, a járatok és foglalások listáját

Ez a dokumentáció a struktúráról a következő fájl alapján készült:

- [Structs.h](#)

5. fejezet

Fájlok dokumentációja

5.1. FlightBook.h fájlreferencia

```
#include "Structs.h"
```

Függvények

- void `fb_load_from_file` (`RuntimeState *pState`)
- void `fb_save_to_file` (`RuntimeState *pState`)
- `FlightBook * fb_alloc_book` ()
- void `fb_free_book` (`FlightBook *book`)
- void `book_flight` (`RuntimeState *pState`)
- void `do_book_flight` (`FlightData *data`, `RuntimeState *pState`)
- void `finalize_book` (`RuntimeState *pState`)

5.1.1. Részletes leírás

Foglalással kapcsolatos műveletek

5.1.2. Függvények dokumentációja

5.1.2.1. `book_flight()`

```
void book_flight (
    RuntimeState * pState )
```

Egz foglalás megvalósításának levezénylése grafikusan

Paraméterek

<i>pState</i>	Futtatási adatok
---------------	------------------

5.1.2.2. do_book_flight()

```
void do_book_flight (
    FlightData * data,
    RuntimeState * pState )
```

Egy foglalás megvalósítása grafikusan

Paraméterek

<i>data</i>	A járat amire foglalni akarunk
<i>pState</i>	Futtatási adatok

5.1.2.3. fb_alloc_book()

```
FlightBook* fb_alloc_book ( )
```

Egz új foglalásstruktúra létrehozása

Visszatérési érték

A létrehozott adatra pointer

5.1.2.4. fb_free_book()

```
void fb_free_book (
    FlightBook * book )
```

Egz foglalás felszabadítása a memóriából

Paraméterek

<i>book</i>	
-------------	--

5.1.2.5. fb_load_from_file()

```
void fb_load_from_file (
    RuntimeState * pState )
```

A mentett foglalások betöltése fájlból

Paraméterek

<i>pState</i>	Futtatási adatok, ide olvassa be a mentett adatoka
---------------	--

5.1.2.6. fb_save_to_file()

```
void fb_save_to_file (
    RuntimeState * pState )
```

Foglalások mentés fájlba

Paraméterek

<i>pState</i>	Futtatási adatok
---------------	------------------

5.1.2.7. finalize_book()

```
void finalize_book (
    RuntimeState * pState )
```

Egy foglalás véglegesítése

Paraméterek

<i>pState</i>	Futtatási adatok
---------------	------------------

5.2. FlightData.h fájlreferencia

```
#include <GenericList.h>
#include "string-utils.h"
#include "Structs.h"
```

Függvények

- `FlightData * fd_alloc_flight ()`

- void `fd_load_from_file` (`RuntimeState` *`pState`)
- void `fd_add_flight` (`RuntimeState` *`pState`, `FlightData` *`data`)
- void `fd_save_to_file` (`RuntimeState` *`pState`)
- void `list_flights` (`RuntimeState` *`pState`)
- void `remove_flight` (`RuntimeState` *`pState`)
- void `find_flight` (`RuntimeState` *`pState`)
- void `new_flight` (`RuntimeState` *`pState`)
- `FlightData` * `find_flight_by_flightnum` (string `fn`, `RuntimeState` *`pState`)

5.2.1. Részletes leírás

Járat adatokalm kapcsolatos műveletek

5.2.2. Függvények dokumentációja

5.2.2.1. `fd_add_flight()`

```
void fd_add_flight (
    RuntimeState * pState,
    FlightData * data )
```

Egy járat hozzáadása a mentett járatokhoz

Paraméterek

<i>pState</i>	A futtáskor tárolandó adatok, ide teszi be az új járatot
<i>data</i>	Az új járat

5.2.2.2. `fd_alloc_flight()`

```
FlightData* fd_alloc_flight ( )
```

Egy járat adatstruktúra lefoglalása

Visszatérési érték

Pointer a struktúrára

5.2.2.3. `fd_load_from_file()`

```
void fd_load_from_file (
    RuntimeState * pState )
```

A mentett járatok adatainak betöltése fájlból

Paraméterek

<i>pState</i>	A futtáskor tárolandó adatok, ide olvassa be a járatokat
---------------	--

5.2.2.4. fd_save_to_file()

```
void fd_save_to_file (
    RuntimeState * pState )
```

A tárolt járatok elmentése

Paraméterek

<i>pState</i>	Futtatási adatok
---------------	------------------

5.2.2.5. find_flight()

```
void find_flight (
    RuntimeState * pState )
```

Egy járat keresésének lebonyolítása grafikusán

Paraméterek

<i>pState</i>	Futtatási adatok
---------------	------------------

5.2.2.6. find_flight_by_flightnum()

```
FlightData* find_flight_by_flightnum (
    string fn,
    RuntimeState * pState )
```

Egz járat megtalálása járatszám alapján

Paraméterek

<i>fn</i>	A keresett járatszám
<i>pState</i>	Futtatási adatok

Visszatérési érték

A megtalált járat vagy NULL

5.2.2.7. list_flights()

```
void list_flights (
    RuntimeState * pState )
```

A járatok kilistázása grafikusán

Paraméterek

<i>pState</i>	Futtatási adatok
---------------	------------------

5.2.2.8. new_flighth()

```
void new_flighth (
    RuntimeState * pState )
```

Egy új járat hozzáadása grafikusán

Paraméterek

<i>pState</i>	
---------------	--

5.2.2.9. remove_flight()

```
void remove_flight (
    RuntimeState * pState )
```

Egy járat törlésének lebonzoltítása grafikusán

Paraméterek

<i>pState</i>	Futtatási adatok
---------------	------------------

5.3. GenericList.h fájlreferencia

```
#include <stdbool.h>
```


Adatszerkezetek

- struct [GenericItem](#)
- struct [GenericList](#)

Típusdefiníciók

- typedef struct [GenericItem](#) [GenericItem](#)
- typedef struct [GenericList](#) [GenericList](#)

Függvények

- [GenericList](#) * [gl_create_list](#) ()
- void [gl_delete_list](#) ([GenericList](#) *list)
- [GenericList](#) * [gl_filter_text](#) ([GenericList](#) *list, char *(*pDataToString)(void *data), char *text, int threshold)

5.3.1. Részletes leírás

Egy generikus (void*) linkelt lista megvalósítás

5.3.2. Típusdefiníciók dokumentációja

5.3.2.1. GenericItem

```
typedef struct GenericItem GenericItem
```

Egy listaelem tárolására szolgáló struktúra Tartalmaz egy értékpointert és egy pointert a következő elemre

5.3.2.2. GenericList

```
typedef struct GenericList GenericList
```

Egy generikus lista tárolására szolgáló struktúra Tartalmaz egy pointert az első elemre és függvénypointereket a lista kezelésére az áttekinthetőség kedvéért

5.3.3. Függvények dokumentációja

5.3.3.1. `gl_create_list()`

```
GenericList* gl_create_list ( )
```

üres lista létrehozása

Visszatérési érték

Pointer a listára

5.3.3.2. `gl_delete_list()`

```
void gl_delete_list (
    GenericList * list )
```

Egy listának és elemeinek felszabadítása

Paraméterek

<i>list</i>	Pointer a listára
-------------	-------------------

5.3.3.3. `gl_filter_text()`

```
GenericList* gl_filter_text (
    GenericList * list,
    char (*)(void *data) pDataToString,
    char * text,
    int threshold )
```

Egy generikus lista eleminek szűrése szöveggé alakítás után egyezési százalék alapján

Paraméterek

<i>list</i>	A lista
<i>pDataToString</i>	Függvénytípus egy elem szöveggé alakítására
<i>text</i>	A referenciaszöveg
<i>threshold</i>	A százaléktérlet-küszöb

Visszatérési érték

Egy új lista a szűrt elemekkel

5.4. `intelligent_menu.h` fájlreferencia

```
#include <stdbool.h>
```

```
#include "string-utils.h"
```

Függvények

- int [varg_options_menu](#) (string question,...)
- int [varg_options_yesno](#) (string question)
- void * [varg_input_menu](#) (string question, bool(*pValidate)(string *, string), void *(*pCreateData)(string *),...)
- int [varg_options_menu_generator](#) (string question, void *data_list, int number_of_items, void *(*pDataItem)(void *data_list, int index), string(*pToString)(void *data))

5.4.1. Részletes leírás

A grafikus menürendszer használatának megkönnyítése Használata opcionális

5.4.2. Függvények dokumentációja

5.4.2.1. varg_input_menu()

```
void* varg_input_menu (
    string question,
    bool(*) (string *, string) pValidate,
    void (*) (string *) pCreateData,
    ... )
```

Egy kérdés és változó szövegbeviteli mező megjelenítés

Paraméterek

<i>question</i>	A kérdés
...	A válasz sztringek, ! az utolsónak NULLnak kell lennie

Figyelmeztetés

Az utolsónak paraméternek NULLnak kell lennie a helyes működéshez

Paraméterek

<i>pValidate</i>	Függvény az adatok hitelesítéséhez aminek első paramétere egy sztring-tömb az adatokkal, a második egy sztring ahova hibaüzenetet lehet írni. A visszatérési érték hiba nélkül true
<i>pCreateData</i>	Függvény a sztringtömbben kapott adatok egy db. void* pontterré alakítására. Pl. struktúrába töltés. A visszatérés a void*

Visszatérési érték

A pCreateData által létrehozott adatpointer

5.4.2.2. varg_options_menu()

```
int varg_options_menu (
    string question,
    ... )
```

Egy kérdés és változó mennyiségű válasz opció megjelenítése

Paraméterek

<i>question</i>	A kérdés
...	A válasz sztringek, ! az utolsónak NULLnak kell lennie

Visszatérési érték

A kiválasztott válasz száma

Figyelmeztetés

Az utolsónak paraméternek NULLnak kell lennie a helyes működéshez

5.4.2.3. varg_options_menu_generator()

```
int varg_options_menu_generator (
    string question,
    void * data_list,
    int number_of_items,
    void (*)(void *data_list, int index) pDataItem,
    string(*) (void *data) pToString )
```

Egy kérdés és egy listából előállított válaszoknak a megjelenítése

Paraméterek

<i>question</i>	A kérdés
-----------------	----------

Visszatérési érték

A kiválasztott válasz száma

Paraméterek

<i>data_list</i>	A lista
<i>number_of_items</i>	A lista hossza
<i>pDataItem</i>	Függvény egy indexedik elem lekérésére
<i>pToString</i>	Függvény egy lekért elem szöveggé alakítására

Visszatérési érték

A kiválasztott elem indexe

5.4.2.4. varg_options_yesno()

```
int varg_options_yesno (
    string question )
```

IgenNem kérdés megjelenítése

Paraméterek

<i>question</i>	A kérdés
-----------------	----------

Visszatérési érték

A válasz 0-igen, 1-nem

5.5. main.c fájlreferencia

```
#include <intelligent_menu.h>
#include <Structs.h>
#include <c-econio/econio.h>
#include <stdlib.h>
#include <FlightData.h>
#include <FlightBook.h>
```

Függvények

- void [win_init](#) ()
- int [main](#) ()

5.5.1. Részletes leírás

Main modul A program fő működését és a menürendszer egy részének funkcióit biztosítja

5.5.2. Függvények dokumentációja

5.5.2.1. main()

```
int main ( )
```

Main belépési pont

Visszatérési érték

5.5.2.2. win_init()

```
void win_init ( )
```

Windowsos működéshez szükséges feladatok elvégzése

5.6. menu.h fájlreferencia

```
#include <stdbool.h>
#include <string-utils.h>
```

Adatszerkezetek

- struct [OptionNode](#)

Típusdefiníciók

- typedef struct [OptionNode](#) [OptionNode](#)

Függvények

- int [menu_options](#) (string question, string *options, int num)
- int [menu_options_list](#) (string question, [OptionNode](#) *listHead)
- void [menu_message](#) (string message)
- string [menu_input](#) (string question)
- void * [menu_input_list](#) (string question, bool(*pfV)(string *, string), void *(*pvC)(string *), string *options, int num)

5.6.1. Részletes leírás

Egy parancssoros grafikus menürendszer megvalósítása

5.6.2. Típusdefiníciók dokumentációja

5.6.2.1. OptionNode

```
typedef struct OptionNode OptionNode
```

Linkelt lista a listás opciómegjelenítéshez

Figyelmeztetés

Belső használatra

5.6.3. Függvények dokumentációja

5.6.3.1. menu_input()

```
string menu_input (  
    string question )
```

Egyszerű kérdés feltétele

Paraméterek

<i>question</i>	A kérdés
-----------------	----------

Visszatérési érték

sztring válasz

5.6.3.2. menu_input_list()

```
void* menu_input_list (  
    string question,  
    bool(*) (string *, string) pfV,  
    void *(*) (string *) pvC,
```

```
string * options,  
int num )
```

Egy kérdés és változó szövegbeviteli mező megjelenítés

Paraméterek

<i>question</i>	A kérdés
<i>options</i>	sztringtömb a mezőnevekkel
<i>num</i>	A mezőnevek száma
<i>pfV</i>	Függvény az adatok hitelesítéséhez aminek első paramétere egy sztring-tömb az adatokkal, a második egy sztring ahova hibaüzenetet lehet írni. A visszatérési érték hiba nélkül true
<i>pvC</i>	Függvény a sztringtömbben kapott adatok egy db. void* pointerre alakítására. Pl. struktúrába töltés. A visszatérés a void*

Visszatérési érték

A pvC által létrehozott adatpointer

5.6.3.3. menu_message()

```
void menu_message (
    string message )
```

Üzenet megjelenítése a felhasználónak

Paraméterek

<i>message</i>	Az üzenet
----------------	-----------

5.6.3.4. menu_options()

```
int menu_options (
    string question,
    string * options,
    int num )
```

Opciólista megjelenítése

Paraméterek

<i>question</i>	Kérdés
<i>options</i>	Sztringtömbben az opciók
<i>num</i>	Az opciók száma

Visszatérési érték

A kiválasztott opció indexe

5.6.3.5. menu_options_list()

```
int menu_options_list (
    string question,
    OptionNode * listHead )
```

Opciólista megjelenítése. Az intelligent_menu használata ajánlott

Paraméterek

<i>question</i>	A kérdés
<i>listHead</i>	Linkelt listás opciók első eleme

Visszatérési érték

A kiválasztott elem indexe

5.7. string-utils.h fájlreferencia

Típusdefiníciók

- typedef char * **string**

Függvények

- int [stra_max](#) (char *strs[], int num)
- string * [allocate_string_array](#) (int itemnum, int item_length)
- void [free_string_array](#) (string *stra, int itemnum)
- void [strtrim](#) (string str)
- int [strcnt](#) (string str, char charToFind)
- string [strrmc](#) (string str, char charToRemove)
- int [streqq](#) (string str, string ref)
- void [strlow](#) (string str)

5.7.1. Részletes leírás

Sztring manipulálási segédfüggvények gyűjteménye Nagyrészen a grafikus menürendszer segédfüggvényei

5.7.2. Függvények dokumentációja

5.7.2.1. allocate_string_array()

```
string* allocate_string_array (
    int itemnum,
    int item_length )
```

Lefoglalja a megfelelő méretű memóriaterületet egy itemnum elemű sztringtömb számára

Paraméterek

<i>itemnum</i>	A tömb eleminek száma
<i>item_length</i>	Egy elem maximális hossza

Visszatérési érték

Pointer a sztringtömbre

5.7.2.2. free_string_array()

```
void free_string_array (
    string * stra,
    int itemnum )
```

Felszabadít egy sztringtömböt és elemeit

Paraméterek

<i>stra</i>	Pointer a tömbre
<i>itemnum</i>	A tömb elemeinek száma

5.7.2.3. stra_max()

```
int stra_max (
    char * strs[],
    int num )
```

Egy sztring-tömbből megmondja a leghosszabb elem hosszát

Paraméterek

<i>strs</i>	Sztring tömb pointer
<i>num</i>	Sztring tömb eleminek száma

Visszatérési érték

A leghosszabb elem hossza

5.7.2.4. `strcnt()`

```
int strcnt (
    string str,
    char charToFind )
```

Megszámolja egy bizonyos karakter előfordulásinak számát egy sztringben

Paraméterek

<i>str</i>	A sztring
<i>charToFind</i>	A karakter

Visszatérési érték

Az előfordulások száma

5.7.2.5. `streqq()`

```
int streqq (
    string str,
    string ref )
```

Kiszámolja két sztring hasonlóságának százalékát

Paraméterek

<i>str</i>	A sztring
<i>ref</i>	A referenciaszöveg

Visszatérési érték

A hasonlóság százaléka

5.7.2.6. `strlow()`

```
void strlow (
    string str )
```

Kicserél minden nagybetűt kisbetűre egy sztringben

Figyelmeztetés

Ez a függvény módosítja a kapott sztringet

Paraméterek

<i>str</i>	A sztring
------------	-----------

5.7.2.7. `strrmc()`

```
string strrmc (
    string str,
    char charToRemove )
```

Eltávolít egy bizonyos karaktert a sztringből

Paraméterek

<i>str</i>	A sztring
<i>charToRemove</i>	Az eltávolítani kívánt karakter

Visszatérési érték

Az új szöveg a karakterek nélkül

5.7.2.8. `strtrm()`

```
void strtrm (
    string str )
```

Szóköz karakterek eltávolítása sztringekből

Paraméterek

<i>str</i>	A sztring
------------	-----------

5.8. Structs.h fájlreferencia

```
#include "string-utils.h"
#include "GenericList.h"
```

Adatszerkezetek

- struct [RuntimeState](#)
- struct [FlightBook](#)
- struct [FlightData](#)

Típusdefiníciók

- typedef enum [Menus](#) [Menus](#)
- typedef struct [RuntimeState](#) [RuntimeState](#)
- typedef struct [FlightBook](#) [FlightBook](#)
- typedef struct [FlightData](#) [FlightData](#)

Enumerációk

- enum [Menus](#) { **NORMAL**, **VEGAN**, **LAKTOZ** }

5.8.1. Részletes leírás

A programhoz szükséges adatstruktúrák

5.8.2. Típusdefiníciók dokumentációja

5.8.2.1. FlightBook

```
typedef struct FlightBook FlightBook
```

Egy foglalás adatait tárol

5.8.2.2. FlightData

```
typedef struct FlightData FlightData
```

Egy járat adatait tárolja

5.8.2.3. Menus

```
typedef enum Menus Menus
```

Menü típusok

5.8.2.4. RuntimeState

```
typedef struct RuntimeState RuntimeState
```

A program futás közbeni állapotát tárolja, a járatok és foglalások listáját

5.8.3. Enumerációk dokumentációja

5.8.3.1. Menus

```
enum Menus
```

Menü típusok

Tárgymutató

allocate_string_array
 string-utils.h, 28

append
 GenericList, 9

book_flight
 FlightBook.h, 13

do_book_flight
 FlightBook.h, 14

fb_alloc_book
 FlightBook.h, 14

fb_free_book
 FlightBook.h, 14

fb_load_from_file
 FlightBook.h, 14

fb_save_to_file
 FlightBook.h, 15

fd_add_flight
 FlightData.h, 16

fd_alloc_flight
 FlightData.h, 16

fd_load_from_file
 FlightData.h, 16

fd_save_to_file
 FlightData.h, 17

finalize_book
 FlightBook.h, 15

find_flight
 FlightData.h, 17

find_flight_by_flightnum
 FlightData.h, 17

FlightBook, 7
 Structs.h, 32

FlightBook.h, 13
 book_flight, 13
 do_book_flight, 14
 fb_alloc_book, 14
 fb_free_book, 14
 fb_load_from_file, 14
 fb_save_to_file, 15
 finalize_book, 15

FlightData, 7
 Structs.h, 32

FlightData.h, 15
 fd_add_flight, 16
 fd_alloc_flight, 16
 fd_load_from_file, 16
 fd_save_to_file, 17

find_flight, 17

find_flight_by_flightnum, 17

list_flights, 18

new_flight, 18

remove_flight, 18

free_string_array
 string-utils.h, 29

GenericItem, 8
 GenericList.h, 19
 next_item, 8
 value, 8

GenericList, 8
 append, 9
 GenericList.h, 19
 get_item, 9
 insert, 10
 remove, 10
 size, 10

GenericList.h, 18
 GenericItem, 19
 GenericList, 19
 gl_create_list, 19
 gl_delete_list, 20
 gl_filter_text, 20

get_item
 GenericList, 9

gl_create_list
 GenericList.h, 19

gl_delete_list
 GenericList.h, 20

gl_filter_text
 GenericList.h, 20

insert
 GenericList, 10

intelligent_menu.h, 20
 varg_input_menu, 21
 varg_options_menu, 22
 varg_options_menu_generator, 22
 varg_options_yesno, 23

list_flights
 FlightData.h, 18

main
 main.c, 24

main.c, 23
 main, 24
 win_init, 24

- menu.h, [24](#)
 - menu_input, [25](#)
 - menu_input_list, [25](#)
 - menu_message, [27](#)
 - menu_options, [27](#)
 - menu_options_list, [27](#)
 - OptionNode, [25](#)
- menu_input
 - menu.h, [25](#)
- menu_input_list
 - menu.h, [25](#)
- menu_message
 - menu.h, [27](#)
- menu_options
 - menu.h, [27](#)
- menu_options_list
 - menu.h, [27](#)
- Menus
 - Structs.h, [32](#)
- new_fligth
 - FlightData.h, [18](#)
- next_item
 - GenericItem, [8](#)
- nextNode
 - OptionNode, [11](#)
- OptionNode, [11](#)
 - menu.h, [25](#)
 - nextNode, [11](#)
 - value, [11](#)
- remove
 - GenericList, [10](#)
- remove_flight
 - FlightData.h, [18](#)
- RuntimeState, [12](#)
 - Structs.h, [32](#)
- size
 - GenericList, [10](#)
- stra_max
 - string-utils.h, [29](#)
- strcnt
 - string-utils.h, [29](#)
- streqq
 - string-utils.h, [30](#)
- string-utils.h, [28](#)
 - allocate_string_array, [28](#)
 - free_string_array, [29](#)
 - stra_max, [29](#)
 - strcnt, [29](#)
 - streqq, [30](#)
 - strlow, [30](#)
 - strrmc, [31](#)
 - strtrm, [31](#)
- strlow
 - string-utils.h, [30](#)
- strrmc
 - string-utils.h, [31](#)
- strtrm
 - string-utils.h, [31](#)
- Structs.h, [31](#)
 - FlightBook, [32](#)
 - FlightData, [32](#)
 - Menus, [32](#)
 - RuntimeState, [32](#)
- value
 - GenericItem, [8](#)
 - OptionNode, [11](#)
- varg_input_menu
 - intelligent_menu.h, [21](#)
- varg_options_menu
 - intelligent_menu.h, [22](#)
- varg_options_menu_generator
 - intelligent_menu.h, [22](#)
- varg_options_yesno
 - intelligent_menu.h, [23](#)
- win_init
 - main.c, [24](#)