Console messenger via fifo

Generated by Doxygen 1.8.13

Contents

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

Chapter 2

File Documentation

2.1 prog.cpp File Reference

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>
```

Functions

- int closeprog (int FILE, char ∗buf)

  Prepares to close the fifo file or closes it before the end of the program.
- int ERROREn (int FILE, char ∗buf)

  Function for an adequate program crash when a user enters an empty message.
- int main ()
- int openfile (char ∗namefile)

  Opens or creates a fifo file using functions mknod () and open ().
- int print (int FILE, char ∗buf, char ∗str, int size_str)

  Writes the text of the message entered by the user to the common message buffer, and also displays the entire buffer on the screen, similar to the show () function.
- int show (int FILE, char ∗buf)

  Displays the entire message buffer on the terminal screen.

Variables

- const int SIZE_BUF_MAX = 1024

  maximum allowed message buffer length.
- const int SIZE_STR_MAX = 256

  terminal messenger.

2.1.1 Function Documentation

2.1.1.1 closeprog()

```
int closeprog (
             int FILE,
             char * buf )
```

Prepares to close the fifo file or closes it before the end of the program.

Parameters

in	*File*	returned by openfile ().
out	*buf*	pointer to the message buffer to write.

Returns

0.

2.1.1.2 ERROREn()

```
int ERROREn (
             int FILE,
             char * buf )
```

Function for an adequate program crash when a user enters an empty message.

Parameters

in	*File*	returned by openfile ().
out	*buf*	pointer to the message buffer to write.

Returns

0.

2.1.1.3 main()

```
int main ( )
```

2.1.1.4 openfile()

```
int openfile (
             char * namefile )
```

Opens or creates a fifo file using functions mknod () and open ().

Parameters

in	*namefile*	full file path.

Returns

int File or error in case the file has not been opened.

2.1.1.5 print()

```
int print (
             int FILE,
             char * buf,
             char * str,
             int size_str )
```

Writes the text of the message entered by the user to the common message buffer, and also displays the entire buffer on the screen, similar to the show () function.

Parameters

in	*File*	returned by openfile ().
out	*buf*	pointer to the message buffer to write.
in	*str*	pointer to the user's message string.
in	*size_str*	message size.

Returns

0.

2.1.1.6 show()

```
int show (
             int FILE,
             char * buf )
```

Displays the entire message buffer on the terminal screen.

Parameters

`in`	*File*	returned by openfile ().
`out`	*buf*	pointer to the message buffer to write.

Returns

0.

2.1.2 Variable Documentation

2.1.2.1 SIZE_BUF_MAX

```
const int SIZE_BUF_MAX = 1024
```

maximum allowed message buffer length.

Constant of the maximum length of the buffer storing user messages.

2.1.2.2 SIZE_STR_MAX

```
const int SIZE_STR_MAX = 256
```

terminal messenger.

Terminal messenger launched in the console of a single computer.maximum allowable length of input string.

Constant of the maximum value of the string length, the message entered by the user.