

mysystem関数の作成

`system`関数のクローン`mysystem`関数を作りなさい。

また、与えられたテストドライバを用いて動作確認をしなさい。

`system`関数は様々なプログラムから呼び出される汎用的なものである。エラーが発生した場合、自分でエラーメッセージを出力せず、エラーの発生を戻り値で呼び出し側に伝えるだけに留めるべきである。

レポジトリの内容

Makefile	makeコマンドが使用するファイル
mysysmain.c	テストドライバ（テスト用のmain関数）
mysystem.c	このファイルにmysystem関数を作成する。

作成

1. system関数の仕様

文法：`int system(char *command);`

説明：`command`で渡された文字列を`/bin/sh`に渡し実行させる。

用途：Cプログラムの中からシェルのコマンドラインと同じ文法でコマンド実行ができる。

詳細：

- `NULL`ポインタを渡された場合は`1`を返す。
- `fork`できなかった場合は`-1`を返す。
- それ以外では`/bin/sh`の終了ステータスを返す。
- `/bin/sh`を起動できない場合、子プロセスは`exit(127);`で終了する。

2. mysystem関数の作成

レポジトリをクローンし`mysystem.c`を編集してプログラムを完成する。

上の仕様を満たすプログラムを美しくコーディングする。

できたと思ったら十分な動作テストを行う。

3. 動作テスト

`mysysmain.c`内に準備してあるmain関数を使用してテストを行う。

実行例

```
% ./mysystem ls -l                                <-- コマンド行引数が多すぎる
使い方 : ./mysysmain コマンド文字列
% ./mysysmain "ls -l README.md"                  <-- 正常にコマンドが実行できる
mysystem:
-rw-r--r--@ 1 sigemura  staff  1966 Jul  7 16:51 README.md
retval = 00000000
system:
-rw-r--r--@ 1 sigemura  staff  1966 Jul  7 16:51 README.md
retval = 00000000
```

```
% ./mysysmain "cat a"          <-- catがエラーを起こす場合
mysystem:
cat: a: No such file or directory
retval = 00000100
system:
cat: a: No such file or directory
retval = 00000100
```

コンパイル方法

Makefileを準備してあるので**make**コマンドでコンパイルできる.

```
% make
cc -Wall -std=c99 -o mysysmain mysysmain.c  mysystem.c
```