



C Piscine

C 11

Summary: このドキュメントは、C Piscine @ 42の C 11モジュール用の課題である。

Contents

I	Instructions	2
II	Foreword	4
III	Exercise 00 : ft_foreach	6
IV	Exercise 01 : ft_map	7
V	Exercise 02 : ft_any	8
VI	Exercise 03 : ft_count_if	9
VII	Exercise 04 : ft_is_sort	10
VIII	Exercise 05 : do-op	11
IX	Exercise 06 : ft_sort_string_tab	13
X	Exercise 07 : ft_advanced_sort_string_tab	14

Chapter I

Instructions

- 課題に関する噂に惑わされないよう気をつけ、信用しないこと。
- この書類は、提出前に変更になる可能性があるため、気をつけること。
- ファイルとディレクトリへの権限があることを、あらかじめ確認すること。
- すべての課題は、提出手順に従い行うこと。
- 課題の確認と評価は、あなたの周りにいるPiscine受験者により行われる。
- 課題の確認と評価は、Piscine受験者に加えて、Moulinetteと呼ばれるプログラムによっても行われる。
- Moulinetteは、大変細かい評価を行う。これはすべて自動で行われるため、交渉の余地はない。
- Moulinetteは、コーディング規範（Norm）を遵守しないコードを解釈することができない。そのため、Moulinetteはnorminetteと呼ばれるプログラムを使用し、あなたのファイルがコーディング規範を遵守しているか確認を行う。せっかくの取り組みが、norminetteの確認により無駄にならないよう、気をつけること。
- 問題は、簡単なものから徐々に難しくなるように並べられている。簡単な問題が解けていない場合は、難しい問題が解けていたとしても 加点されることはない。
- 使用が禁止されている関数を使用した場合は、不正とみなされる。不正者は-42の評価をつけられ、この評価に対する交渉の余地はない。
- 課題がプログラムの提出を要求する際は、main()関数のみを提出すること。
- Moulinetteは以下のフラグを用いて、gccでコンパイルする。 -Wall -Wextra -Werror
- プログラムがコンパイルされなかった場合、評価は0になる。
- 課題で指定されていないものは、どんなファイルもディレクトリ内に置かないこと。

- 質問がある場合は、隣の人に聞くこと。それでも分からない場合は、反対側の席の人に聞くこと。
- 助けてくれるのは、Google / 人間 / インターネット / ...と呼ばれているものたちである。
- 出力例には、問題文に明記されていない細部まで表示されている場合があるため、入念に確認すること。

Chapter II

Foreword

Here's a little story :

(1982, California) Larry Walters of Los Angeles is one of the few to contend for the Darwin Awards and live to tell the tale. "I have fulfilled my 20-year dream," said Walters, a former truck driver for a company that makes TV commercials. "I'm staying on the ground. I've proved the thing works."

Larry's boyhood dream was to fly. But fates conspired to keep him from his dream. He joined the Air Force, but his poor eyesight disqualified him from the job of pilot. After he was discharged from the military, he sat in his backyard watching jets fly overhead.

He hatched his weather balloon scheme while sitting outside in his "extremely comfortable" Sears lawnchair. He purchased 45 weather balloons from an Army-Navy surplus store, tied them to his tethered lawnchair (dubbed the Inspiration I) and filled the four-foot diameter balloons with helium. Then, armed with some sandwiches, Miller Lite, and a pellet gun, he strapped himself into his lawnchair. He figured he would shoot to pop a few of the many balloons when it was time to descend.

Larry planned to sever the anchor and lazily float to a height of about 30 feet above the backyard, where he would enjoy a few hours of flight before coming back down. But things didn't work out quite as Larry planned.

When his friends cut the cord anchoring the lawnchair to his Jeep, he did not float lazily up to 30 feet. Instead he streaked into the LA sky as if shot from a cannon, pulled by the lift of 45 helium balloons, holding 33 cubic feet of helium each.

He didn't level off at 100 feet, nor did he level off at 1000 feet. After climbing and climbing, he leveled off at 16,000 feet.

At that height he felt he couldn't risk shooting any of the balloons, lest he unbalance the load and really find himself in trouble. So he stayed there, drifting cold and frightened with his beer and sandwiches, for more than 14 hours. He crossed the primary approach corridor of LAX, where startled Trans World Airlines and Delta Airlines pilots radioed in reports

of the strange sight.


Eventually he gathered the nerve to shoot a few balloons, and slowly descended. The hanging tethers tangled and caught in a power line, blacking out a Long Beach neighborhood for 20 minutes. Larry climbed to safety, where he was arrested by waiting members of the LAPD. As he was led away in handcuffs, a reporter dispatched to cover the daring rescue asked him why he had done it. Larry replied nonchalantly, "A man can't just sit around."

The Federal Aviation Administration was not amused. Safety Inspector Neal Savoy said, "We know he broke some part of the Federal Aviation Act, and as soon as we decide which part it is, a charge will be filed."

The moral of this story is Larry Walters should have stay on his chair and learn C....

Chapter III

Exercise 00 : ft_foreach

	Exercise 00
	ft_foreach
	提出するディレクトリ : <i>ex00/</i>
	提出するファイル : <i>ft_foreach.c</i>
	使用可能な関数 : None

- 与えられた配列にある各int型を、関数ポインタfに渡す関数ft_foreachを作成せよ。
- 関数ポインタには、配列の順で値が渡される。
- プロトタイプ例)


```
void ft_foreach(int *tab, int length, void(*f)(int));
```

- 以下のような関数ft_foreachを呼び出すと、int型の配列にあるすべての要素が標準出力に出力される。
- 例)

```
ft_foreach(tab, 1337, &ft_putnbr);
```

Chapter IV

Exercise 01 : ft_map


	Exercise 01
	ft_map
	提出するディレクトリ : <i>ex01/</i>
	提出するファイル : ft_map.c
	使用可能な関数 : malloc

- 与えられた配列にある各int型を、関数ポインタfに渡し、すべての戻り値を保持している配列を返す関数ft_mapを作成せよ。
- 関数ポインタには、配列の順で値が渡される。
- プロトタイプ例)

```
int      *ft_map(int *tab, int length, int(*f)(int));
```


Chapter V

Exercise 02 : ft_any

	Exercise 02
	ft_any
提出するディレクトリ : <i>ex02/</i>	
提出するファイル : <i>ft_any.c</i>	
使用可能な関数 : None	


- 配列の各要素を、関数ポインタfに与え、その戻り値が0ではない場合が1つ以上ある場合は1を返し、それ以外の場合は0を返す、ft_any関数を作成せよ。
- 関数ポインタには、配列の順で値が渡される。
- プロトタイプ例)

```
int ft_any(char **tab, int(*f)(char*));
```

- 配列の終端はNULLポインタである。

Chapter VI

Exercise 03 : ft_count_if


	Exercise 03
	ft_count_if
	提出するディレクトリ : <i>ex03/</i>
	提出するファイル : <i>ft_count_if.c</i>
	使用可能な関数 : None

- 配列の各要素を関数ポインタfに与え、0以外を返した回数を返す関数ft_count_ifを作成せよ。
- 関数ポインタには、配列の順で値が渡される。
- プロトタイプ例)

```
int ft_count_if(char **tab, int length, int(*f)(char*));
```

Chapter VII

Exercise 04 : ft_is_sort


	Exercise 04
	ft_is_sort
	提出するディレクトリ : <i>ex04/</i>
	提出するファイル : <i>ft_is_sort.c</i>
	使用可能な関数 : None

- 配列が昇順に並んでいる場合は1を返し、それ以外の場合は0を返す関数ft_is_sortを作成せよ。
- 引数として与えられている関数は、1番目の引数が2番目の引数より小さい場合は、負の整数を返し、2番目の引数と等しい場合は0を返し、それ以外の場合は正の整数を返す。
- プロトタイプ例)

```
int ft_is_sort(int *tab, int length, int(*f)(int, int));
```

Chapter VIII

Exercise 05 : do-op

	Exercise 05
do-op	
提出するディレクトリ : <i>ex05/</i>	
提出するファイル : プログラムを作成するために必要なファイル	
使用可能な関数 : <code>write</code>	

- do-opと呼ばれるプログラムを作成せよ。
- このプログラムは、3つのコマンドライン引数を受け取り実行する。: do-op value1 operator value2
- 例)

```
$> ./do-op 42 "+" 21
63
$>
```

- 演算子に対応する関数を、関数ポインタの配列から選択し、使用すること。
- 無効な演算子である場合、プログラムは0を標準出力に出力すること。
- コマンドライン引数の数が正しくない場合、do-opは何も出力しないこと。
- プログラムは演算子として次の記号を使用し、その結果を出力すること。: “+”、“-”、“/”、“*”、“%”
- 割られる数が0である場合は、以下のように出力すること。

```
Stop : division by zero
```

- 割る数が0である場合は、以下のように出力すること。


```
Stop : modulo by zero
```

- Moulinetteのテストの例)

```
$> make clean
$> make
$> ./do-op
$> ./do-op 1 + 1
2
$> ./do-op 42amis - ---20toto12
62
$> ./do-op 1 p 1
0
$> ./do-op 1 + toto3
1
$>
$> ./do-op toto3 + 4
4
$> ./do-op foo plus bar
0
$> ./do-op 25 / 0
Stop : division by zero
$> ./do-op 25 % 0
Stop : modulo by zero
$>
```

Chapter IX

Exercise 06 : ft_sort_string_tab


	Exercise 06
	ft_sort_string_tab
	提出するディレクトリ : <i>ex06/</i>
	提出するファイル : <i>ft_sort_string_tab.c</i>
	使用可能な関数 : None

- tab内の文字列をASCII順に並べ替える関数ft_sort_string_tabを作成せよ。
- tabの終端はNULLポインタである。
- 配列内のポインタを交換することにより、並べ替えが実行される。
- プロトタイプ例)

```
void ft_sort_string_tab(char **tab);
```

Chapter X

Exercise 07 : ft_advanced_sort_string_tab

	Exercise 07
ft_advanced_sort_string_tab	
提出するディレクトリ : ex07/	
提出するファイル : ft_advanced_sort_string_tab.c	
使用可能な関数 : None	

- 配列にある各文字列を関数ポインタcmpに与え、その戻り値に応じて配列を並べ替える関数ft_advanced_sort_string_tabを作成せよ。
- 配列内のポインタを交換することにより、並べ替えが実行される。
- tabの終端はNULLポインタである。
- プロトタイプ例)

```
void ft_advanced_sort_string_tab(char **tab, int(*cmp)(char *, char *));
```



ft_advanced_sort_string_tab()に渡す関数ポインタとしてft_strcmpを指定すると、ft_sort_string_tab()と同じ結果を返す。