



C Piscine

C 00

Summary: このドキュメントはC Piscine @ 42の C 00モジュール用の課題です。

Contents

I	Instructions	2
II	Foreword	4
III	Exercice 00 : ft_putchar	5
IV	Exercise 01 : ft_print_alphabet	6
V	Exercise 02 : ft_print_reverse_alphabet	7
VI	Exercise 03 : ft_print_numbers	8
VII	Exercise 04 : ft_is_negative	9
VIII	Exercise 05 : ft_print_comb	10
IX	Exercise 06 : ft_print_comb2	11
X	Exercise 07 : ft_putnbr	12
XI	Exercise 08 : ft_print_combn	13

Chapter I

Instructions

- このページのみを参考にしてください。噂を信用しないで下さい。
- この書類は、提出前に変更になる可能性があります。十分に注意して下さい。
- 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して下さい。
- 課題は全て提出手順に従って行って下さい。
- 課題の確認と評価は、あなたのクラスメイトが行います。
- 課題はMoulinetteと呼ばれるプログラムによっても確認・評価されます。
- Moulinetteは大変細かい評価を行います。全て自動で行われ、交渉方法はありません。頑張ってください。
- Moulinetteは規範を無視したコードは解読できません。Moulinetteはあなたのファイルが規範を遵守しているかをチェックするために、`norminette`と呼ばれるプログラムを使って判断します。要約：せっかくの取り組みが`norminette`のチェックによって無駄になるのは勿体無いので、気をつけましょう。
- 課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ます。簡単な課題が解けていない場合、難しい問題かが解けていたとしても **加点されることはありません**。
- 禁止されている関数をしようした場合は不正とみなします。不正者は-42の評価をつけられこの評価に交渉の余地はありません。
- プログラムを要求する際は`main()`関数のみを提出しましょう。
- Moulinetteはこれらのフラグを用いてgccでコンパイルします：-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です。
- 課題で指定されているもの以外はどんなファイルもディレクトリ内に残しておくことはできません。
- 質問があれば右側の人に聞きましょう。それでも分からなければ左側の人に聞いてください。

- あなたを助けてくれるのはGoogle / 人間 / インターネット / ...と呼ばれているものです。
- intranet上のフォーラムの” C Piscine” パートかPiscineのslackを確認してください。
- 例を徹底的に調べてください。課題で言及されていない詳細まで要求されます。
- By Odin, by Thor! 頭を使って!!!



Norminetteは、 `-R CheckForbiddenSourceHeader` をオプションに追加しなければなりません。その際、Moulinetteも使用します。

Chapter II

Foreword

タラ肝油は、タラ魚の肝臓に由来する栄養補助食品です。

ほとんどの魚油と同様に、オメガ-3脂肪酸、
エイコサペンタエン酸（EPA）およびドコサヘキサエン酸（DHA）が高レベルで含まれています。

タラ肝油には、ビタミンAとビタミンDも含まれています。


歴史的には、ビタミンAとビタミンDの含有量のために摂取されてきました。

ビタミンDはくる病やその他のビタミンD欠乏症の症状を予防することが示されているため、
かつては子供に一般的に投与されていました。

タラ肝油とは反対に、Cは身体にいいです。積極的に摂取しましょう！

Chapter III

Exercice 00 : ft_putchar

	Exercise 00
	ft_putchar
	提出するディレクトリ : <i>ex00/</i>
	提出するファイル : <i>ft_putchar.c</i>
	使用可能な関数 : <i>write</i>

- パラメータとして渡された文字を標準出力する関数を書きましょう。
- プロトタイプ例)


```
void ft_putchar(char c);
```

文字を表示させるためには、次のようにwrite関数を使う必要があります。

```
write(1, &c, 1);
```

Chapter IV

Exercise 01 : ft_print_alphabet


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

- 小文字のアルファベットを「a」から昇順に1行で表示する関数を作成しましょう。
- プロトタイプ例)

```
void ft_print_alphabet(void);
```

Chapter V

Exercise 02 : ft_print_reverse_alphabet


	Exercise 02
	ft_print_reverse_alphabet
	提出するディレクトリ : <i>ex02/</i>
	提出するファイル : <i>ft_print_reverse_alphabet.c</i>
	使用可能な関数 : <i>write</i>

- 小文字のアルファベットを「z」から降順に1行で表示する関数を作成しましょう。
- プロトタイプ例)

```
void ft_print_reverse_alphabet(void);
```


Chapter VI

Exercise 03 : ft_print_numbers


	Exercise 03
	ft_print_numbers
	提出するディレクトリ : <i>ex03/</i>
	提出するファイル : <i>ft_print_numbers.c</i>
	使用可能な関数 : <i>write</i>

- 全数字を昇順に1行で表示する関数を作成しましょう。
- プロトタイプ例)

```
void ft_print_numbers(void);
```

Chapter VII

Exercise 04 : ft_is_negative


	Exercise 04
	ft_is_negative
	提出するディレクトリ : <i>ex04/</i>
	提出するファイル : <i>ft_is_negative.c</i>
	使用可能な関数 : write

- パラメーターとして入力された符号付き整数に応じて'N'か'P'を標準出力する関数を作成しましょう。 *n*がマイナスの場合は'N'、*n*がプラスかnullの場合、'P'を標準出力します。
- プロトタイプ例)

```
void ft_is_negative(int n);
```

Chapter VIII

Exercise 05 : ft_print_comb

	Exercise 05
	ft_print_comb
	提出するディレクトリ : <i>ex05/</i>
	提出するファイル : <i>ft_print_comb.c</i>
	使用可能な関数 : <i>write</i>

- 3桁の数字の起こりうる全ての組み合わせを昇順で表示する関数を作成しましょう。 -繰り返すかは任意です。
- 出力例)


```
$> ./a.out | cat -e
012, 013, 014, 015, 016, 017, 018, 019, 023, ..., 789$>
```

- 789が既に存在するので987は存在しません。
- 9が1回以上存在するので999は存在しません。
- プロトタイプ例)

```
void ft_print_comb(void);
```

Chapter IX

Exercise 06 : ft_print_comb2

	Exercise 06
	ft_print_comb2
	提出するディレクトリ : <i>ex06/</i>
	提出するファイル : <i>ft_print_comb2.c</i>
	使用可能な関数 : <i>write</i>

- 00-99までの2桁の数字の異なる全ての組み合わせを昇順で表示する関数を作成しましょう。
- 出力例)


```
$>./a.out | cat -e
00 01, 00 02, 00 03, 00 04, 00 05, ..., 00 99, 01 02, ..., 97 99, 98 99$>
```

- プロトタイプ例)

```
void ft_print_comb2(void);
```

Chapter X

Exercise 07 : ft_putnbr

	Exercise 07
	ft_putnbr
	提出するディレクトリ : <i>ex07/</i>
	提出するファイル : <i>ft_putnbr.c</i>
	使用可能な関数 : <i>write</i>


- パラメータとして渡された数を標準出力する関数を作成しましょう。その際、`int`型に収まる全てのパターンを標準出力できるようにしてください。
- プロトタイプ例)

```
void ft_putnbr(int nb);
```

- 例)
 - `ft_putnbr(42)` は "42" と標準出力する。

Chapter XI

Exercise 08 : ft_print_combn

	Exercise 08
	ft_print_combn
	提出するディレクトリ : <i>ex08/</i>
	提出するファイル : <i>ft_print_combn.c</i>
	使用可能な関数 : <i>write</i>

- *n*の数の異なる組み合わせの全てを昇順で表示する関数を作成しましょう。
- *n*は次のようになるでしょう: $0 < n < 10$
- *n* = 2の場合の出力例)

```
$> ./a.out | cat -e
01, 02, 03, ..., 09, 12, ..., 79, 89$>
```

- プロトタイプ例)

```
void ft_print_combn(int n);
```