



C Piscine

C 07

Summary: このドキュメントは、C Piscine @ 42の C 07モジュール用の課題である。

Contents

I	Instructions	2
II	Foreword	4
III	Exercise 00 : ft_strdup	6
IV	Exercise 01 : ft_range	7
V	Exercise 02 : ft_ultimate_range	8
VI	Exercice 03 : ft_strjoin	9
VII	Exercise 04 : ft_convert_base	10
VIII	Exercise 05 : ft_split	11

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。



norminetteは、 `-R CheckForbiddenSourceHeader` をオプションに追加しなければならない。Moulinetteも、このオプションを使用する。

Chapter II

Foreword

Morty: Rick!

Rick: Uhp-uhp-uhp! Morty, keep your hands off your ding-dong! It's the only way we can speak freely. Look around you, Morty. Do you really think this wuh-world is real? You'd have to be an idiot not to notice all the sloppy details. Look, that guy's putting a bun between two hot dogs.

Morty: I dunno, Rick, I mean, I've seen people do that before.

Rick: Well, look at that old lady. She's-she's walking a cat on a leash.

Morty: Uh, Mrs. Spencer does that all the time, Rick.

Rick: Look, I-I-I don't want to hear about Mrs. Spencer, Morty! She's an idiot! All right, all right, there. Wh-what about that, Morty?

Morty: Okay, okay, you got me on that one.

Rick: Oh, really, Morty? Are you sure you haven't seen that somewhere in real life before?

Morty: No, no, I haven't seen that. I mean, why would a Pop-Tart want to live inside a toaster, Rick? I mean, th-that would be like the scariest place for them to live. Y'know what I mean?

Rick: You're missing the point, Morty. Why would he drive a smaller toaster with wheels? I mean, does your car look like a smaller version of your house? No.

Morty: So, why are they doing this? W-what do they want?

Rick: Well, that would be obvious to you, Morty, if you'd been paying attention. [an ambulance drives past Rick and Morty and stops; open back doors]

Paramedic: We got the President of the United States in here! We need 10cc of concentrated dark matter, stat, or he'll die!

Morty: Concentrated dark matter? They were asking about that in class.

Rick: Yeah, it's a special fuel I invented to travel through space faster than anybody else. These Zigerions are always trying to scam me out of my secrets, but they made a big mistake this time, Morty. They dragged you into this. Now they're gonna pay!


Morty: What do you— w-w-what are we gonna do?

Rick: We're gonna scam the scammers, Morty. And we're gonna take 'em for everything they've got.

The following exercises will be easier to complete if you are a fan of "Rick and Morty"

Chapter III

Exercise 00 : ft_strdup


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

- `strdup` (man `strdup`)関数の挙動を再現してください。
- プロトタイプ例)

```
char *ft_strdup(char *src);
```

Chapter IV

Exercise 01 : ft_range

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


- int型の配列を返す関数ft_rangeを作成せよ。
- このint型の配列に、minとmaxの間にあるすべての値（minの値を含み、maxの値は除く）が格納されていること。
- プロトタイプ例)

```
int *ft_range(int min, int max);
```

- minの値がmaxの値より大きい、または、同じである場合は、NULLを返すこと。

Chapter V

Exercise 02 : ft_ultimate_range

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


- 動的にメモリを確保し、`int`型の配列を格納し、その配列のサイズを返す関数`ft_ultimate_range`を作成せよ。この`int`型の配列に、`min`と`max`の間にあるすべての値（`min`の値を含み、`max`の値は除く）が格納されていること。
- プロトタイプ例)

```
int ft_ultimate_range(int **range, int min, int max);
```

- 関数が`range`のサイズを返すように実装すること。エラーの場合は、-1を返すこと。
- `min`の値が`max`の値より大きい、または、同じである場合は、`range` にNULLを指定し、0を返すこと。

Chapter VI

Exercice 03 : ft_strjoin


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

- *sep*を各文字列の間に挟み、*strs*配列にあるすべての文字列を連結する関数を作成せよ。
- *size*は、*strs*配列の文字列の数である。
- *size*が0である場合、メモリの解放が可能な空の文字列を返すこと。
- プロトタイプ例)

```
char *ft_strjoin(int size, char **strs, char *sep);
```

Chapter VII

Exercise 04 : ft_convert_base


	Exercise 04
	ft_convert_base
	提出するディレクトリ : <i>ex04/</i>
	提出するファイル : <i>ft_convert_base.c, ft_convert_base2.c</i>
	使用可能な関数 : <i>malloc, free</i>

- nbr文字列を、base_from進数からbase_to進数へ変換した文字列を返す関数を作成せよ。
- nbr、base_from、base_toに、書き込みはできない。
- nbrは、ft_atoi_base と同じルールが適用される。 C 07モジュールを参照し、“+”、“-”、空白の扱いに気をつけること。
- nbrは、intに格納されること。
- ベースにエラーがある場合は、NULLを返すこと。
- 空白や“+”を、戻り値の前に入れないこと。必要な場合は、“-”を1つのみ入れること。
- プロトタイプ例)

```
char *ft_convert_base(char *nbr, char *base_from, char *base_to);
```

Chapter VIII

Exercise 05 : ft_split

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

- 文字列にある各文字に応じて、文字列を分割する関数を作成せよ。
- *charset*文字列にある各文字は、*str*文字列を分割するために使用すること。
- この関数は、*charset*文字列にある文字が*str*文字列を分割し、分割された文字列の配列を返す。
- 文字列の配列のメモリのみではなく、分割された各文字列のためにメモリを確保し、コピーすること。
- 配列に空の文字列を含めることはできない。
- 引数として与えられた文字列は、編集することはできない。
- プロトタイプ例)

```
char **ft_split(char *str, char *charset);
```