



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す。

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

- 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はこれらのフラッグを用いて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。



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

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 exercices 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>

- Reproduce the behavior of the function `strdup` (man `strdup`).
- 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`を作成しましょう。なお、この`int`型の配列は、`min`と`max`の間の全ての値を含みます。
- `min`の値を含み、`max`の値を除きます。
- プロトタイプ例

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

- もし`min`の値が`max`の値より大きい場合は、`null`を返します。

Chapter V

Exercise 02 : ft_ultimate_range

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


- int型の配列の割り当てを返す関数ft_ultimate_rangeを作成しましょう。なお、このint型の配列は、minとmaxの間の全ての値を含みます。
- minの値を含み、maxの値を除きます。
- プロトタイプ例

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

- rangeのサイズを返します（もしくはエラーの場合、-1を返します）。
- もしminの値がmaxの値より大きい場合は、range がnullを指し、0を返します。

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ながら`strs`配列にあるすべての文字列を連結する関数を作成しましょう。
- `size`は`strs`配列の文字列の数です。
- もし`size`が0であれば、メモリの解放が可能な空の文字列を返します。
- 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 : ex04/
	提出するファイル : ft_convert_base.c, ft_convert_base2.c
	使用可能な関数 : malloc, free

- nbr文字列をbase_from進数からbase_to進数へと変換した文字列を返す関数を作成しましょう。
- nbr, base_from, base_toは書き込み不可です。
- nbrはft_atoi_base (別のモジュールを参照)と同じルールが適用されます。その際、 '+'、 '-'と空白に気をつけましょう。
- nbrはintに必ず収まります。
- ベースに誤りがある場合は、 NULLを返します。
- 空白や '+' を返り値の接頭辞に入れないでください。必要な場合は '-' を一つのみ入れてください。
- 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>

- 文字列の各文字に応じて、ある文字列を分ける関数を作成しましょう。
- *charset*文字列にある各文字は、*str*文字列を分けるために使用しましょう。
- この関数は、*charset*文字列にある文字が*str*文字列を分断し、分断された文字列の配列を返します。
- 文字列の配列のメモリ確保はもちろんのこと、分断された各文字列用にメモリを確保しコピーしてください。
- 配列内に空の文字列を置くことはできません。適宜、結論を見つけ出してください。
- 引数として与えられた文字列は修正不可です。
- プロトタイプ例

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