



C Piscine

C 07

Staff 42 [pedago@42.fr](mailto:pedago@42.fr)

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

*Version: 6*

# Contents

<b>I</b>	<b>Instructions</b>	<b>2</b>
<b>II</b>	<b>AI Instructions</b>	<b>4</b>
<b>III</b>	<b>Foreword</b>	<b>7</b>
<b>IV</b>	<b>Exercise 00 : ft_strdup</b>	<b>9</b>
<b>V</b>	<b>Exercise 01 : ft_range</b>	<b>10</b>
<b>VI</b>	<b>Exercise 02 : ft_ultimate_range</b>	<b>11</b>
<b>VII</b>	<b>Exercise 03 : ft_strjoin</b>	<b>12</b>
<b>VIII</b>	<b>Exercise 04 : ft_convert_base</b>	<b>13</b>
<b>IX</b>	<b>Exercise 05 : ft_split</b>	<b>14</b>
<b>X</b>	<b>Submission and peer-evaluation</b>	<b>15</b>

# Chapter I

## Instructions

- このページのみが正式な課題の指示です。噂に惑わされないようにしてください。
- この課題は提出前に変更される可能性がありますので、気をつけてください！
- ファイルとディレクトリへの適切な権限があることを確認してください。
- すべての課題で、提出手順に従ってください。
- 提出した課題の評価（レビュー）は、あなたの周りにはいるPiscine受験者によって行われます。
- さらに、**Moulinette**（自動採点プログラム）にも評価されます。
- **Moulinette**は非常に厳格かつ厳密に評価を行います。完全に自動化されており、交渉の余地はありません。思わぬ減点を避けるためにも、細部まで丁寧に取り組んでください。
- **Moulinette**は融通が利きません。あなたのコードがNormに準拠していない場合、理解しようとしません。**Moulinette**は**norminette**というプログラムを使用して、ファイルがNormに準拠しているかチェックします。要約：**norminette**のチェックに通らない課題を提出しても意味がありません。
- 課題は難易度が低いものから高いものへと難易度順に並んでいます。前の課題が正しく解けていない場合、後の課題が解けていても評価されません。
- 禁止されている関数を使用することは不正行為とみなされます。不正行為者は**-42**という成績がつけられ、これは交渉の余地がありません。
- 私たちが具体的にプログラムを求める場合のみ、**main()**関数を提出する必要があります。
- **Moulinette**は以下のフラグでコンパイルします：**-Wall -Wextra -Werror, cc**を使用します。
- プログラムがコンパイルできない場合、**0**の評価になります。
- 課題で指定されたファイル以外は、ディレクトリに絶対に残さないでください。

- 質問がある場合は、まず右隣の仲間に聞いてみましょう。いなければ左隣の仲間に聞いてみてください。
- あなたの助けになる参考資料はGoogle / man / インターネット / ...です。
- Intranetのフォーラムの「C Piscine」セクションをチェックしてください。
- 課題の例（出力例）をよく読んでください。課題の説明文だけではすぐに分からない要件が隠れていることがあります。
- 頭を使ってください!!!



Norminetteは、`-R CheckForbiddenSourceHeader` をオプションに追加しなければなりません。Moulinetteも同様にこのオプションを使用します。

# Chapter II

## AI Instructions

### ● Context

C Piscineは強烈な体験です。42で迎える最初の大きな挑戦として、問題解決、自律性、そしてコミュニティの世界に深く飛び込むことになります。

このフェーズの主な目的は、試行錯誤や反復、そして特にピアラーニングを通じた交流によって、自分の土台を築き上げることです。

AI時代において、近道を見つけることは簡単です。しかし、AIの利用が本当にあなたの成長を助けているのか、それとも真のスキルを身につける上で障害となっていないか、よく考えることが大切です。

Piscineはまたとない人との繋がりを経験する場でもあります。今のところ、それにとって代わるものは何にもありません。AIでさえも。

学習ツールとして、ICTカリキュラムの一環として、そして労働市場での期待の高まりに応えるものとして、私たちのAIに対する考え方の詳細については、Intranet上の専用FAQをご覧ください。

### ● 主なメッセージ

- ➡ 近道をせず、強固な基礎を築く。
- ➡ 技術力と実践力を真に養う。
- ➡ 真のピアラーニングを体験し、学び方や新たな問題の解決方法を学び始める。
- ➡ 結果よりも学習の過程が重要である。
- ➡ AIに伴うリスクを学び、一般的な落とし穴を避けるための効果的な管理方法と対策を身につける。

## ● 学習者のルール:

- 課題には、AIに頼る前にまず自分の頭で考えること。
- AIに直接的な答えを求めないこと。
- AIに対する42のグローバルな方針を学ぶこと。

## ● このフェーズでの到達目標:

この基礎フェーズを終えることで、以下の点を達成できます。

- 技術とコーディングの適切な基礎を身につける。
- この段階においてAIが危険となりうる理由と、その危険性を理解する。

## ● コメントと具体例:

- 私たちはAIの存在を知っていますし、それがあなたのプロジェクトを解決できることも知っています。しかし、あなたは学ぶためにここにいるのであって、AIが学習したことを証明するためではありません。AIが特定の問題を解決できることを示すためだけに、あなた（や私たち）の時間を無駄にしないでください。
- 42での学習は、答えを知ることではありません。答えを見つけ出す能力を養うことです。AIは直接答えを与えてくれますが、それはあなた自身の論理的思考を妨げます。そして、論理的思考には時間と努力、そして失敗が伴います。成功への道は、決して楽なものではありません。
- Exam中はAIが利用できないことを心に留めておいてください。インターネットもスマートフォンもありません。学習過程でAIに過度に頼りすぎている場合、すぐにそのことに気づくでしょう。
- ピアラーニングは、多様な考え方やアプローチに触れる機会を与え、対人スキルや多角的な思考能力を向上させます。これは、単にボットとチャットするよりもはるかに価値があります。ですから、恥ずかしがらずに、話しかけ、質問し、共に学びましょう！
- AIは学習ツールとして、またそれ自体がトピックとして、カリキュラムの一部になります。独自のAIソフトウェアを開発する機会さえあります。私たちの段階的なアプローチについてさらに学ぶためには、Intranetで利用可能なドキュメントを参照してください。

### ✓ 良い実践例:

新しいコンセプトの学習でつまづいています。近くの人にどうアプローチしたか尋ねてみます。10分ほど話していると、突然ひらめき、理解できました。

**✕ 悪い実践例:**

こっそりAIを使い、良さそうなコードをコピーします。ピアレビューの際には何も説明できず、不合格になりました。ExamではAIがなく、また行き詰まり、不合格になりました。

# Chapter III

## 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 IV

## 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関数の動作を再現しなさい。(man strdup)
- プロトタイプ例 :

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

# Chapter V

## 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`を作成してください。この`int`型の配列には、`min`以上 `max`未満のすべての値が格納されている必要があります。
- `min`の値は含み、`max`の値は除いてください。
- プロトタイプ例 :

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

- `min`の値が`max`以上の場合、または同じである場合は、`NULL`を返してください。

# Chapter VI

## 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`を作成してください。この`int`型の配列には、`min`以上`max`未満のすべての値が格納されている必要があります。
- `min`の値は含み、`max`の値は除いてください。
- 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してください。エラーの場合は、`-1`を返すようにしてください。
- `textttmin`の値が`max`以上の場合、`range`は`NULL`を指すようにし、戻り値として`0`を返してください。

# Chapter VII

## Exercise 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してください。
- `size`は、`strs`配列の文字列の数です。
- `size`が0の場合、`free()`を適用することが可能な空の文字列を返してください。
- プロトタイプ例 :

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

# Chapter VIII

## 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>

- 文字列`nbr`を、`base_from`数から`base_to`数に変換した結果を返す関数を作成してください。
- `nbr`、`base_from`、`base_to`に書き込みはできません。
- `nbr`は、別のモジュールの`ft_atoi_base`と同じルールが適用されます。“+”、“-”、空白文字に注意してください。
- `nbr`の数値は、`int`型に格納される必要があります。
- ベースが無効な場合は、`NULL`を返してください。
- 必要な場合に限り、戻り値の先頭に“-”を1つのみ入れてください。空白や“+”は戻り値の前に入れないでください。
- プロトタイプ例 :

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

# Chapter IX

## 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 : <b>ft_split.c</b>
	使用可能な関数 : <b>malloc</b>

- 追加で与えられた文字列を区切り文字として使い、文字列を分割する関数を作成してください。
- `charset`文字列にある各文字を、区切り文字として使用する必要があります。
- 関数は配列を返す必要があります。配列の各要素には、2つの区切り文字の間にある文字列のアドレスが格納されている必要があります。配列の最後の要素はNULLにして、配列の終わりを示してください。
- 配列内に空文字列を含めることはできません。これを踏まえて適切な処理方法を考えてください。
- 引数として与えられる文字列は変更できません。
- プロトタイプ例 :

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

# Chapter X

## Submission and peer-evaluation

課題は、いつも通りGitリポジトリに提出してください。リポジトリ内の提出物のみが、レビュー中の評価対象となります。ファイル名が正しいかどうか、必ず確認してください。



この課題の指示で、明示的に求められたファイルのみを提出してください。