



C Piscine

C 00

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

*Version:* 8

# 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_putchar</b>	<b>8</b>
<b>V</b>	<b>Exercise 01 : ft_print_alphabet</b>	<b>9</b>
<b>VI</b>	<b>Exercise 02 : ft_print_reverse_alphabet</b>	<b>10</b>
<b>VII</b>	<b>Exercise 03 : ft_print_numbers</b>	<b>11</b>
<b>VIII</b>	<b>Exercise 04 : ft_is_negative</b>	<b>12</b>
<b>IX</b>	<b>Exercise 05 : ft_print_comb</b>	<b>14</b>
<b>X</b>	<b>Exercise 06 : ft_print_comb2</b>	<b>15</b>
<b>XI</b>	<b>Exercise 07 : ft_putnbr</b>	<b>16</b>
<b>XII</b>	<b>Exercise 08 : ft_print_combn</b>	<b>17</b>
<b>XIII</b>	<b>Submission and peer-evaluation</b>	<b>18</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!!!



すべての.cおよび.hファイルに、標準42ヘッダーを必ず含めてください。  
Norminetteは必ずその有無を確認します！



Norminetteは、`-R CheckForbiddenSourceHeader` をオプションに追加しなければなりません。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

タラ肝油は、タラ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）、ドコサヘキサエン酸（DHA）が豊富に含まれてい  
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の補給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。

タラ肝油とは違い、Cはとても良い味です。積極的に摂取しましょう！



# Chapter IV

## Exercise 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に出力する関数を作成してください。
- プロトタイプ例 :

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

- 文字を標準出力に出力するためには、以下のように**write**関数を使用してください。


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



初めて提出する場合、課題を再提出できるようになるまでの時間は短いので、進捗を確認するため、躊躇わずにレビューを一度受けてみましょう！

# Chapter V

## 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すべてを、'a'から昇順に1行で標準出力に出力する関数を作成してください。
- プロトタイプ例 :


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



周りにいる誰かに、気軽に質問してみましょう！

# Chapter VI

## 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すべてを、'z'から降順に1行で標準出力に出力する関数を作成してください。
- プロトタイプ例 :


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



こまめにGitをプッシュしましょう！

# Chapter VII

## 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>

- すべての数字を、昇順に1行で標準出力に出力する関数を作成してください。
- プロトタイプ例 :


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



協力し合うことが成功の鍵です。

# Chapter VIII

## 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 : <b>ft_is_negative.c</b>
	使用可能な関数 : <b>write</b>

- パラメータとして渡された整数に応じて、'N'または、'P'を標準出力に出力する関数を作成してください。
  - **n**がマイナスの場合は、'N'を標準出力に出力してください。
  - **n**がプラスまたは0の場合は、'P'を標準出力に出力してください。
- プロトタイプ例 :

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



失敗も学びの一部です。

## マイルストーン達成、引き続き頑張りましょう！

この課題の必須問題をすべて完了しました。ここから先、引き続きこの課題の任意問題に取り組むか、次の課題に進むかを選ぶことができます。


- 任意問題に進んでさらに学ぶ。
- 次の課題に進む。

どちらの道に進んでも、役立つ概念を学べます。下記の点を踏まえて判断してください。

- 最初の試験（Exam）や週末に行われるチーム課題（Rush）はC言語が中心です。そのため、事前に学んで経験を積んでおくといいでしょう。
- Piscineの成績は、1つの要素だけではなく様々な観点から評価されます。
  - 各課題の完了状況
  - Piscineの課題全体の進捗度最善の結果につながるように、賢く選択しましょう。
- Piscineが終了するまでの間、同じ課題は、数日から数週間後に再び取り組むことができます。
- 他のPiscine受験生とある程度同じ進捗を保つことで、より円滑に学び合うことができます。

# Chapter IX

## 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つの異なる数字による重複のない組み合わせをすべて、各組み合わせ内の数字と組み合わせ自体の両方が昇順になるように標準出力に出力する関数を作成してください。

- 出力例 :

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

- 789が既に存在するため、987は含まれません。
- 999のように、同じ数字が複数回現れる組み合わせも含まれません。
- プロトタイプ例 :


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



隣のPiscine受験生に確認してみましたか？

# Chapter X

## 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>

- 2桁の異なる2つの数字（00から99まで）のすべての組み合わせ（XX XX）を、昇順で標準出力に出力する関数を作成してください。

- 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);
```




他のPiscine受験生からヒントを得ても、自分の手を動かして作成しましょう！



# Chapter XI

## 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に出力する関数を作成してください。  
この関数は、**int**型に収まるすべてのパターンを標準出力に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を標準出力に出力します。



どんな情報でも鵜呑みにせず、必ず自分でテスト・検証・確認をしましょう！

# Chapter XII

## 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**個の異なる数字による重複のない組み合わせをすべて、昇順で標準出力に出力する関数を作成してください。
- **n**の値は次の条件を満たします :  $0 < n < 10$
- 例 : **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);
```



隣のPiscine受験生に確認してみましたか？

# Chapter XIII

## 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。