



C Piscine

Shell 00

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

Version: 5.0

Contents

I	Instructions	2
II	AI Instructions	3
III	Foreword	6
IV	Exercise 00 : Z	8
V	Exercise 01 : testShell00	9
VI	Exercise 02 : Oh yeah, mooore...	10
VII	Exercise 03 : SSH me!	11
VIII	Exercise 04 : midLS	12
IX	Exercise 05 : GiT commit	13
X	Exercise 06 : gitignore	15
XI	Exercise 07 : diff	16
XII	Exercise 08 : clean	17
XIII	Exercise 09 : Illusions, not tricks, Michael...	18
XIV	Submission and peer-evaluation	19

Chapter I

Instructions

- 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。
- ファイルとディレクトリへの権限があることを事前に確認してください。
- 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**という自動評価プログラムでもチェック・採点されます。
- **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。
- すべてのシェルの課題は、必ず/bin/shで実行できるようにする必要があります。
- 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。
- 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。

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ともありません。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答えるものとして、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

Below are the lyrics to *City Hunter*'s theme song, "*Moonlight Shadow*":

The last time ever she saw him
Carried away by a moonlight shadow
He passed on worried and warning
Carried away by a moonlight shadow.
Lost in a riddle that Saturday night
Far away on the other side.
He was caught in the middle of a desperate fight
And she couldn't find how to push through

The trees that whisper in the evening
Carried away by a moonlight shadow
Sing a song of sorrow and grieving
Carried away by a moonlight shadow
All she saw was a silhouette of a gun
Far away on the other side.
He was shot six times by a man on the run
And she couldn't find how to push through

[Chorus]
I stay, I pray
See you in Heaven far away...
I stay, I pray
See you in Heaven one day.

Four A.M. in the morning
Carried away by a moonlight shadow
I watched your vision forming
Carried away by a moonlight shadow
A star was glowing in the silvery night
Far away on the other side
Will you come to talk to me this night
But she couldn't find how to push through


[Chorus]

Far away on the other side.
Caught in the middle of a hundred and five
The night was heavy and the air was alive
But she couldn't find how to push through
Carried away by a moonlight shadow
Carried away by a moonlight shadow
Far away on the other side.

Unfortunately, this topic has nothing to do with City Hunter.

Chapter IV

Exercise 00 : Z

	Exercise 00
Only the best know how to display Z	
提出するディレクトリ : <i>ex00/</i>	
提出するファイル : <i>z</i>	
使用可能な関数 : <i>None</i>	

- `cat`というコマンドが使用されるたびに“Z”と改行を返す、`z`というファイルを作成してください。


```
?>cat z
Z
?>
```



Googleはあなたの友達です。

Chapter V

Exercise 01 : testShell00

	Exercise 01
What are attributes anyway ?	
提出するディレクトリ : <i>ex01/</i>	
提出するファイル : <i>testShell00.tar</i>	
使用可能な関数 : None	

- 提出用ディレクトリ内にtestShell00というファイルを作成してください。
- 下記のような出力結果（"total 1"の行は除く）になるように工夫してください。

```
%> ls -l
total 1
-r--r-xr-x 1 XX XX 40 Jun 1 23:42 testShell00
%>
```

- 完了したら、下記のコマンドを実行して提出用ファイルを作成してください。

```
%> tar -cf testShell00.tar testShell00
```




- “XX” の部分に別の内容が表示されていても問題ありません。
- タイムスタンプの“時刻”の部分が“年”で表示されていても問題ありません。



隣のPiscine受験生に確認してみましたか？

Chapter VI

Exercise 02 : Oh yeah, mooore...

	Exercise 02
Oh yeah, mooore...	
提出するディレクトリ : <i>ex02/</i>	
提出するファイル : <i>exo2.tar</i>	
使用可能な関数 : None	

- 以下のファイルとディレクトリを作成し、`ls -l`コマンドを実行したときに下記のような出力結果になるようにしてください。

```
%> ls -l
total XX
drwx--xr-x 2 XX XX  XX Jun 1 20:47 test0
-rwx--xr-- 1 XX XX  4 Jun 1 21:46 test1
dr-x---r-- 2 XX XX  XX Jun 1 22:45 test2
-r-----r-- 2 XX XX  1 Jun 1 23:44 test3
-rw-r----x 1 XX XX  2 Jun 1 23:43 test4
-r-----r-- 2 XX XX  1 Jun 1 23:44 test5
lrwxrwxrwx 1 XX XX  5 Jun 1 22:20 test6 -> test0
%>
```

- 完了したら、下記のコマンド実行して提出用ファイルを作成してください。

```
%> tar -cf exo2.tar *
```




- “XX” の部分に別の内容が表示されていても問題ありません。
- タイムスタンプの“時刻”の部分が“年”で表示されていても問題ありません。



周りにいる誰かに、気軽に質問してみましょう！

Chapter VII

Exercise 03 : SSH me!

	Exercise 03
SSHキー	
提出するディレクトリ : <i>ex03/</i>	
提出するファイル : <i>id_rsa_pub</i>	
使用可能な関数 : None	

- 自分のSSH鍵を作成してください。作成したら、以下の手順を行ってください。
 - 公開鍵の内容を*id_rsa_pub*というファイルの中にコピーして提出してください。
 - SSH鍵をIntranet上で更新してください。これにより、リポジトリをGitのサーバーに提出できるようになります。




- ファイル名は意図的にこの名前になっています。
- 公開鍵と秘密鍵の違いを理解しておきましょう。



隣のPiscine受験生に確認してみましたか？

Chapter VIII

Exercise 04 : midLS

	Exercise 04
	midLS
	提出するディレクトリ : <i>ex04/</i>
	提出するファイル : <i>midLS</i>
	使用可能な関数 : <i>None</i>

- midLSというファイルの中に、現在のディレクトリ内のファイルとディレクトリ（1つのドットで始まる隠しファイルや、2つのドットで始まるファイルを除く）を一覧表示するコマンドを書いてください。
- 出力は更新日時順に並び、各ファイルやディレクトリはカンマとスペースで区切ってください。
- ディレクトリ名の末尾にはスラッシュ（/）を付けてください。




課題の指示で求められていること以外は、行わないでください。



- RTFM!
- こまめにGitをプッシュしましょう。

Chapter IX

Exercise 05 : GiT commit

	Exercise 05
GiT commit?	
提出するディレクトリ : <i>ex05/</i>	
提出するファイル : <i>git_commit.sh</i>	
使用可能な関数 : None	

- あなたのGitリポジトリ内で、直近5件のコミットIDを出力するシェルスクリプトを作成してください。

```
%> bash git_commit.sh | cat -e
baa23b54f0adb7bf42623d6d0a6ed4587e11412a$
2f52d74b1387fa80eea844969e8dc5483b531ac1$
905f53d98656771334f53f59bb984fc29774701f$
5ddc8474f4f15b3fcb72d08fcb333e19c3a27078$
e94d0b448c03ec633f16d84d63beaef9ae7e7be8$
%>
```

- このスクリプトは42独自の環境下でテストされます。



- RTFM!
- 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！

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 X

Exercise 06 : gitignore

	Exercise 06
GiT	
提出するディレクトリ : <i>ex06/</i>	
提出するファイル : <i>git_ignore.sh</i>	
使用可能な関数 : None	

- Gitリポジトリに反映されないファイルを、すべてリストアップする短いシェルスクリプトを作成してください。

出力例 :

```
%> bash git_ignore.sh | cat -e
.DS_Store$
mywork.c~$
%>
```


- このスクリプトは42独自の環境下でテストされます。



- RTFM!
- 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 : diff

	Exercise 07
提出するディレクトリ : <i>ex07/</i>	
提出するファイル : <i>b</i>	
使用可能な関数 : <i>None</i>	

- 出力が以下のようになる、*b*というファイルを作成してください。

```
%>cat -e a
STARWARS$
Episode IV, A NEW HOPE It is a period of civil war.$
$
Rebel spaceships, striking from a hidden base, have won their first victory against the evil
Galactic Empire.$
During the battle, Rebel spies managed to steal secret plans to the Empire's ultimate weapon
, the DEATH STAR,$
an armored space station with enough power to destroy an entire planet.$
$
Pursued by the Empire's sinister agents, Princess Leia races home aboard her starship,
custodian of the stolen plans that can save her people and restore freedom to the
galaxy...$
$
```


```
%>diff a b > sw.diff
```



- `man patch`
- 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 : clean

	Exercise 08
提出するディレクトリ : <i>ex08/</i>	
提出するファイル : <i>clean</i>	
使用可能な関数 : <i>None</i>	


- *clean*というファイルに、以下の処理を行うコマンドを1つ書いてください。
 - 現在のディレクトリとそのサブディレクトリの中から、*~*で終わるファイル、または*#*で始まり*#*で終わるファイルを検索する。
 - 見つかったファイルを出力し、その後に削除する。
- 使用できるコマンドは1つのみです。“*;*”や“*&&*”などのコマンドの連結は禁止です。



- *man find*
- 協力し合うことが成功の鍵です！

Chapter XIII

Exercise 09 : Illusions, not tricks, Michael...

	Exercise 09
Illusions, not tricks, Michael...	
提出するディレクトリ : <code>ex09/</code>	
提出するファイル : <code>ft_magic</code>	
使用可能な関数 : <code>None</code>	

- `ft_magic`という**magic**ファイルを作成してください。このファイルは、`file`コマンドで、42バイト目に"42"の文字列があるファイルを42 fileというファイルタイプとして検出できるようにしてください。



- `man file`
- 失敗も学びの一部です！

Chapter XIV

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。