



C Piscine

Rush 00

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

Contents

I	Instructions	2
II	Foreword	4
III	Main subject	6
IV	Rush 00	8
V	Rush 01	10
VI	Rush 02	11
VII	Rush 03	12
VIII	Rush 04	13
IX	Submission and peer-evaluation	14

Chapter I

Instructions

- チームは、自動的にレビューに登録される。
- 2回目のレビューは実施されないため、キャンセルしないこと。
- 課題に関して、むやみに詳細な説明を求めることは、課題を複雑にする。
- すべての課題は、提出手順に従い行うこと。
- この課題は、提出一時間前までに変更される可能性がある。
- プログラムは以下のフラグを用いて、cc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。✓
- プログラムはNormに準拠して作成する必要がある。bonus用のファイルや関数がある場合、それらはノームチェックに含まれ、ノームエラーがある場合の評価は0になる。
- エラーを正しく処理する必要がある。エラーになる場合は、エラーメッセージを出力したり、ユーザに制御を戻すこと。
- Rushは、2〜4人のチームで取り組む課題である。
- チームにつけるRush番号は、次のルールに従い設定すること。: チームリーダーのログイン名の頭文字のアルファベット順のインデックス (1〜26) に、5を剰余演算 (modulo 5) した結果の番号をRush番号とする。
- 指定されたチームのメンバーと、必ず協同して課題に取り組み、レビューを受ける際は、チームのメンバー全員が出席すること。
- 課題は、レビューを受ける際までに完了させること。チームのメンバーが、取り組んだ課題について詳細な説明を行うために、レビューは実施される。
- チームのメンバー全員が、課題の目的を十分に理解していることが求められる。課題を分担して取り組んだ場合を含め、課題の全容について、全員が理解していることを確認すること。レビューでは、最も不十分であったと判断された説明を基準に、チームの取り組みが評価される。

- チームを結束させることに對し、チームリーダーのみではなく、一人ひとりのメンバーにその責務がある。電話やメールなど、メンバーと連絡を取る手段を共有すること。このような行動を怠った場合に、弁解する余地はない。
- あらゆる手段を試した場合でも、チームのメンバーと連絡が取れない際は、引き続き目の前の課題に取り組むこと。どのように対処をするか、42のスタッフが判断を行う。また、チームリーダーのみではなく、メンバー全員に提出ディレクトリへアクセスする権限がある。
- ほかの課題にも取り組んだり、コマンドライン引数を持ちいたプログラムを提出することで、ボーナスポイントを獲得できる。



ほかの課題（ボーナス）に取り組む前に、チームに課されている必須の課題が完璧に行えているかを確認すること。必須の課題が解けていない場合は、ほかの課題が解けていたとしても、評価は0になる。

Chapter II

Foreword

Here's the lyrics of a famous TV show for everyone :

[Verse 1]

I wanna be the very best
Like no one ever was
To catch them is my real test
To train them is my cause

I will travel across the land
Searching far and wide
Each pokemon to understand
The power that's inside

[Chorus]

Pokemon! Gotta catch 'em all! It's you and me
I know it's my destiny,
Pokemon! Oh you're my best friend
In a world we must defend
Pokemon! A heart so true
Our courage will pull us through,

You teach me and I'll teach you,
Pokemon! Gotta catch'em all

[Chorus]


Every challenge along the way
With courage I will face.
I will battle every day
To claim my rightful place.
Come with me,
The time is right,
There's no better team.
Arm in arm we'll win the fight!
It's always been our dream!

[Chorus]

I could bet you were singing right now, but it doesn't matter for the moment. And this subject is not related with Pocket Monster by the way...

Chapter III

Main subject

	Exercise 00
Rush0X	
提出するディレクトリ : <i>ex00/</i>	
提出するファイル : <i>main.c, ft_putchar.c, rush0X.c</i>	
使用可能な関数 : <i>write</i>	

- 提出するファイル: *main.c, ft_putchar.c, rush0X.c* (“0X” はRush番号を表す。例 : *rush00.c*)
- それらのファイルを一緒にコンパイルされる。
- *ft_putchar*関数を*ft_putchar.c*ファイルに組み込むこと。
- *main.c*の例)

```
int main()
{
    rush(5, 5);
    return (0);
}
```

- *x, y*と呼ばれる、2つの*int*型の値を引数とする関数*rush*を作成せよ。*rush0X.c*ファイルに組み込むこと。
- 横の長さが*x*、高さが*y*の長方形を標準出力に出力すること。
- 関数がクラッシュしたり、無限ループを行わないこと。
- レビュー中に、すべての関数が適切に処理できているかをチェックするために、*main*の内容は上書きされる。テストの実行例)

```
int     main()
{
    rush(123, 42);
    return (0);
}
```


Chapter IV

Rush 00

- rush(5,3) の表示例

```
$> ./a.out  
o---o  
|   |  
o---o  
$>
```

- rush(5, 1) の表示例

```
$> ./a.out  
o---o  
$>
```

- rush(1, 1) の表示例

```
$> ./a.out  
o  
$>
```

- rush(1, 5) の表示例

```
$> ./a.out  
o  
|  
|  
|  
o  
$>
```

- `rush(4, 4)` の表示例

```
$> ./a.out  
o--o  
|  |  
|  |  
o--o  
$>
```

Chapter V

Rush 01

- rush(5,3) の表示例

```
$> ./a.out  
/***\  
*  *  
\**/  
$>
```

- rush(5, 1) の表示例

```
$> ./a.out  
/***\  
$>
```

- rush(1, 1) の表示例

```
$> ./a.out  
/  
$>
```

- rush(1, 5) の表示例

```
$> ./a.out  
/  
*  
*  
*  
\  
$>
```

- rush(4, 4) の表示例

```
$> ./a.out  
/**\  
*  *  
*  *  
\**/  
$>
```

Chapter VI

Rush 02

- `rush(5,3)` の表示例

```
$> ./a.out
ABBBA
B  B
CBBBC
$>
```

- `rush(5, 1)` の表示例

```
$> ./a.out
ABBBA
$>
```

- `rush(1, 1)` の表示例

```
$> ./a.out
A
$>
```

- `rush(1, 5)` の表示例

```
$> ./a.out
A
B
B
B
C
$>
```

- `rush(4, 4)` の表示例

```
$> ./a.out
ABBA
B  B
B  B
CBBC
$>
```

Chapter VII

Rush 03

- rush(5,3) の表示例

```
$> ./a.out  
ABBBC  
B B  
ABBBC  
$>
```

- rush(5, 1) の表示例

```
$> ./a.out  
ABBBC  
$>
```

- rush(1, 1) の表示例

```
$> ./a.out  
A  
$>
```

- rush(1, 5) の表示例

```
$> ./a.out  
A  
B  
B  
B  
A  
$>
```

- rush(4,4) の表示例

```
$> ./a.out  
ABBC  
B B  
B B  
ABBC  
$>
```

Chapter VIII

Rush 04

- rush(5,3) の表示例

```
$> ./a.out  
ABBBC  
B  B  
CBBBA  
$>
```

- rush(5, 1) の表示例

```
$> ./a.out  
ABBBC  
$>
```

- rush(1, 1) の表示例

```
$> ./a.out  
A  
$>
```

- rush(1, 5) の表示例

```
$> ./a.out  
A  
B  
B  
B  
C  
$>
```

- rush(4, 4) の表示例

```
$> ./a.out  
ABBC  
B  B  
B  B  
CBBA  
$>
```

Chapter IX

Submission and peer-evaluation

課題は、いつも通り Git リポジトリに提出すること。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る。ファイルの名前が正しいことを確認すること。



この課題の要件で求められているファイルのみを提出すること。