C Piscine

Rush 00

Staff 42 pedago@42.fr

Summary: This document is the subject for Rush00 of the C Piscine @ 42.

Contents

Chapter 1

Instructions

- Each member of the group can register the whole group to the evaluation.
- The group MUST be registered to an evaluation.
- Be mindful of the submission procedures indicated at the start of every exercise.
- Moulinette compiles with these flags: **-Wall -Wextra -Werror**, and uses **gcc**.
- If your program doesn't compile, it will be graded 0.
- Rush projects have to be carried out by group of 2, 3 or 4.
- To determine which rush exercise your group will have to complete, follow the instructions in the **groups_s_subject.txt** file.
- You must complete the project with the imposed team and show up at the evaluation slot you've selected, with all of your teammates.
- You project must be done by the time you get to the evaluation. The purpose of evaluation is for you to present and explain any and all details of your work.
- Each member of your group must be fully aware of the inner workings of the project. Should you choose to split the workload, make sure you all understand what everybody has done. During evaluation, you'll be asked questions, and the final grade will be based on the worst explanations.
- Gathering the group for work and evaluation is your responsibility.
- If you want bonus points, you may submit other subjects.
- Moulinette relies on **norminette** to check if your files respect the Norm. An exercise containing files that do not respect the Norm will be graded 0.

Make sure the subject that was originally assigned to your group works perfectly before considering bonuses : If a bonus subject works, but the original one fails the tests, you'll get 0.

`norminette` must be launched with the `-R CheckForbiddenSourceHeader` flag. Moulinette will use it too.

Chapter 2

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 3

Main subject

Turn-in directory : ex00/

Files to turn in: main.c, ft_putchar.c, rush0X.c

Allowed functions: write

- Files to submit: **main.c**, **ft_putchar.c** and your **rush0X.c**, '0X' represents the rush number. For example **rush00.c**.
- Example of main.c :

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

- You must therefore create the function **rush** taking two variables of type **int** as arguments, named respectively **x** and **y**.
- Your function **rush** should display (on-screen) a rectangle of **x** characters for width, and **y** characters for length.
- Your **main** will be modified during evaluation, to check if you've handled everything you're supposed to. Here's an example of test we'll perform :

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

Chapter 4

Rush 00

- rush(5,3) should display :

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

- rush(5, 1) should display :

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

- rush(1, 1) should display :

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

- rush(1, 5) should display :

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

- rush(4, 4) should display :

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

Chapter 5

Rush 01

- rush(5,3) should display :

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

- rush(5, 1) should display :

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

- rush(1, 1) should display :

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

- rush(1, 5) should display :

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

- rush(4, 4) should display :

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

Chapter 6

Rush 02

- rush(5,3) should display :

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

- rush(5, 1) should display :

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

- rush(1, 1) should display :

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

- rush(1, 5) should display :

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

- rush(4, 4) should display :

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

Chapter 7

Rush 03

- rush(5,3) should display :

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

- rush(5, 1) should display :

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

- rush(1, 1) should display :

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

- rush(1, 5) should display :

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

- rush(4, 4) should display :

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

Chapter 8

Rush 04

- rush(5,3) should display :

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

- rush(5, 1) should display :

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

- rush(1, 1) should display :

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

- rush(1, 5) should display :

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

- rush(4, 4) should display :

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