



# Piscina C

## Rush 00

*Sumário: Este documento é o tema do Rush00 da Piscina C da 42.*

*Versão: 5.2*

# Conteúdo

<b>I</b>	<b>Instruções</b>	<b>2</b>
<b>II</b>	<b>Preâmbulo</b>	<b>4</b>
<b>III</b>	<b>Tema Comum</b>	<b>6</b>
<b>IV</b>	<b>Rush 00</b>	<b>8</b>
<b>V</b>	<b>Rush 01</b>	<b>10</b>
<b>VI</b>	<b>Rush 02</b>	<b>11</b>
<b>VII</b>	<b>Rush 03</b>	<b>12</b>
<b>VIII</b>	<b>Rush 04</b>	<b>13</b>
<b>IX</b>	<b>Entrega e avaliação entre pares</b>	<b>14</b>

# Capítulo I

## Instruções

- O grupo será inscrito para a avaliação automaticamente.
- Não cancele a avaliação, você não terá uma outra.
- Qualquer pedido de esclarecimento sobre um dos temas complicará o tema.
- Você deve seguir o procedimento de entrega para todos os seus exercícios.
- O tema pode mudar até uma hora antes da entrega.
- O programa deverá compilar com as flags: -Wall -Wextra -Werror, e utiliza cc.
- Se o seu programa não compila, você terá 0.
- Seu programa deverá ser escrito de acordo com a Norma. Se você tiver funções/arquivos bônus, eles serão incluídos na verificação de norma e você receberá um 0 se eles tiverem algum erro.
- Você deverá lidar com os erros de forma coerente. Sinta-se à vontade para imprimir uma mensagem de erro, ou simplesmente retornar o controle para o usuário.
- Os exercícios do rush devem ser feitos em grupos de 2, 3 ou 4.
- O número de rush imposto para o seu grupo seguirá esta regra:  
ordem alfabética da primeira letra do login do líder da equipe (de 1 a 26) módulo 5.
- Você deverá, portanto, fazer o projeto com o time que foi imposto e estar presente no seu slot de avaliação, com todos os seus parceiros.
- Seu projeto deve ser feito a tempo de fazer a avaliação. O propósito da avaliação é para que você apresente e explique seu trabalho.
- Cada membro do grupo deve estar totalmente ciente do conteúdo do trabalho. Se vocês decidirem dividir o trabalho, assegure-se de que todos entendam o que todos

fizeram. Durante a avaliação, você responderá perguntas e a nota final será baseada nas piores explicações.

- Obviamente, reunir o grupo é de sua responsabilidade. Vocês têm todos os meios para entrar em contato com seus parceiros: telefone, email, pombo-correio, bola de cristal, etc. Nem precisa inventar desculpas. A vida não é justa, ela só é como é.
- Se, depois de realmente tentar de tudo, um de seus parceiros ainda estiver incomunicável: faça o seu rush e arranjaremos uma solução na defesa. Mesmo se foi o líder do grupo: vocês todos possuem acesso ao repositório.
- Se você quiser pontos de bônus, você pode entregar outros temas ou usar argumentos de programa para testar sua função.



Tenha certeza de que o tema que foi designado originalmente para o seu grupo funcione perfeitamente antes de considerar fazer os bônus: Se um bônus funciona, mas o tema original falha nos testes, você terá um 0.

# Capítulo II

## Preâmbulo

Aqui vai a letra de um tema de um programa de TV para todos:

Esse meu jeito de viver  
Ninguém nunca foi igual  
A minha vida é fazer  
O bem vencer o mal

Pelo mundo viajarei  
Tentando encontrar  
Um Pokémon e com o seu poder  
Tudo transformar

(Pokémon! Temos que pegar)  
Isso eu sei  
Pegá-los eu tentarei! (Pokémon!)  
Juntos teremos que  
O mundo defender!

Pokémon! (Temos que pegá-los)  
Isso eu sei  
Pegá-los eu tentarei  
Vai ser grande a emoção  
Pokémon!

Temos que pegar  
Temos que pegar!

Desafios vou encontrar  
E os enfrentarei  
Lutando pelo meu lugar  
Todo dia estarei!

Vem comigo, vamos formar  
Sempre a melhor equipe  
E sempre juntos vamos vencer  
O sonho é poder

Pokémon! (Temos que pegá-los)  
Isso eu sei  
Pegá-los eu tentarei! (Pokémon!)  
Juntos teremos que  
O mundo defender!

Pokémon! (Temos que pegá-los)  
Isso eu sei  
Pegá-los eu tentarei!  
Vai ser grande a emoção! Pokémon!  
(Temos que pegar)  
Temos que pegar

Temos que pegar  
Temos que pegar

Pokémon! (Temos que pegá-los)  
Isso eu sei  
Pegá-los eu tentarei! (Pokémon!)  
Juntos teremos que  
O mundo defender!


Pokémon! (Temos que pegá-los)  
Isso eu sei  
Pegá-los eu tentarei!  
Vai ser grande a emoção!  
Pokémon!  
Temos que pegar  
(Temos que pegar)  
Temos que pegar

Pokémon!

Eu aposto que você estava cantando agora mesmo, mas não importa para o momento.  
E esse tema não está relacionado com Pokémon, inclusive.

# Capítulo III

## Tema Comum

	Exercício : 00
	rush0X
	Pasta de entrega : <i>ex00/</i>
	Ficheiros para entregar : <i>main.c</i> , <i>ft_putchar.c</i> , <i>rush0X.c</i>
	Funções autorizadas : <i>write</i>

- Os arquivos que devem ser entregues são o *main.c*, um *ft\_putchar.c* e o seu *rush0X.c*, no qual 0X corresponde ao número do rush. Por exemplo, *rush00.c*.
- Esses arquivos devem ser compilados juntos.
- Seu arquivo *ft\_putchar.c* deve incluir a função *ft\_putchar*.
- Exemplo de *main.c*:

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

- Você deverá, portanto, escrever a função **rush** tendo como parâmetro duas variáveis de tipo *int* nomeadas respectivamente *x* e *y*. Obviamente esta função deverá estar no arquivo *rush0X.c*.
- Sua função **rush** deverá exibir na tela um retângulo de *x* caracteres de largura, e *y* caracteres de altura.
- Sua função não deve dar crash ou loop infinito.
- A sua *main* será modificada na avaliação, para verificar se você lidou com tudo o que deveria corretamente. Aqui um exemplo de teste que nós faremos:

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



# Capítulo IV

## Rush 00

- `rush(5,3)` exhibirá isto:

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

- `rush(5, 1)` isto:

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

- `rush(1, 1)` isto :

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

- `rush(1, 5)` isto :

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

- `rush(4, 4)` isto :

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

# Capítulo V

## Rush 01

- `rush(5,3)` exhibirá isto:

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

- `rush(5, 1)` isto :

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

- `rush(1, 1)` isto :

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

- `rush(1, 5)` isto :

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

- `rush(4, 4)`, isto:

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

# Capítulo VI

## Rush 02

- `rush(5,3)` exhibirá isto:

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

- `rush(5, 1)`, isto:

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

- `rush(1, 1)` ceci :

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

- `rush(1, 5)`, isto:

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

- `rush(4, 4)`, isto:

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

# Capítulo VII

## Rush 03

- `rush(5,3)` exhibirá isto:

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

- `rush(5, 1)` isto :

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

- `rush(1, 1)` isto :

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

- `rush(1, 5)`, isto:

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

- `rush(4, 4)`, isto:

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

# Capítulo VIII

## Rush 04

- `rush(5,3)` exhibirá isto:

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

- `rush(5, 1)`, isto:

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

- `rush(1, 1)` isto :

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

- `rush(1, 5)`, isto:

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

- `rush(4, 4)`, isto:

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

# Capítulo IX

## Entrega e avaliação entre pares

Entregue seu trabalho no seu repositório `Git` como de costume. Somente o trabalho que estiver dentro do repositório será analisado durante a avaliação. Não hesite em verificar mais uma vez os nomes dos seus arquivos para ter certeza de que estão certos.

Como esses exercícios não são verificados por um programa, fique à vontade para organizar os arquivos como quiser, desde que você tenha entregado os arquivos mandatórios e preencham os requisitos.



Você deve entregar somente os arquivos requisitados pelo tema deste projeto.