



C Piscine

Rush 00

*Summary:* 이 문서는 42 C Piscine 과정 Rush 00 모듈의 학습 주제입니다.

# Contents

|      |              |    |
|------|--------------|----|
| I    | Instructions | 2  |
| II   | Foreword     | 4  |
| III  | Main subject | 5  |
| IV   | Rush 00      | 6  |
| V    | Rush 01      | 8  |
| VI   | Rush 02      | 9  |
| VII  | Rush 03      | 10 |
| VIII | Rush 04      | 11 |

# Chapter I

## Instructions

- 그룹의 각 구성원이 그룹 전체를 코렉션에 등록할 수 있습니다.
- 그룹은 반드시 디펜스에 등록되어야 합니다.
- 과제와 관한 의문은 과제를 더욱 복잡하게 만들 수 있습니다.
- 모든 과제에 대해 과제물 제출 절차를 따라야 합니다.
- 이 문서는 제출 전 최대 1시간 전까지 바뀔 수 있습니다.
- 과제는 나이도에 따라 쉬운 과제에서 어려운 과제 순으로 짜여 있습니다. 앞 단계 과제의 과제물이 완벽하게 작동하지 않으면 나이도가 더 높은 과제는 아무리 잘 완료했다 하더라도 평가에 반영되지 않습니다.
- Moulinette는 -Wall -Wextra -Werror 플래그를 지정하여 컴파일하며 gcc를 사용합니다.
- 프로그램이 컴파일되지 않으면 0점 처리됩니다.
- Rush 과제는 2명, 3명 또는 4명이 한 그룹을 이루어 완료해야 합니다.
- 팀에서 의무적으로 해야 하는 rush 과제의 수는 다음 규칙에 따라 결정합니다.: 팀 리더의 로그인 아이디의 첫 번째 글자의 알파벳 인덱스(1~26)를 5로 나눈 나머지 (mod 5).
- 따라서 프로젝트를 할당된 팀과 함께 하고, 선택한 코렉션 시간에 모든 팀 구성원과 함께 참석해야 합니다.
- 코렉션 일정까지 프로젝트가 완료되어야 합니다. 코렉션의 목적은 여러분이 각자 작업의 모든 세부적인 내용을 발표하고 설명하는 것입니다.
- 그룹의 각 구성원은 프로젝트의 작업 내용에 대해 완벽하게 이해하고 있어야 합니다. 과제를 나누어 하기로 할 경우에는 모든 구성원이 서로 각자 작업 내용을 이해하고 있도록 해야 합니다. 코렉션 중에는 질문을 받게 되며, 최종 점수는 코렉션에서 설명을 가장 잘 못한 것을 기준으로 하여 채점됩니다.
- 굳이 말할 필요도 없겠지만, 그룹에서 빠지는 사람이 없도록 쟁기는 것도 여러분의 책임입니다. 전화, 이메일, 비둘기, 텔레파시 등 팀 구성원에게 연락할 수 있는 모든 수단을 사용하세요. 변명 해봤자 소용없습니다. 인생이란 게 원래 늘 공정하지는 않으니까요.

- 하지만 정말 온갖 방법을 다 동원해 봤지만 연락이 안되는 팀 구성원이 있다면, 프로젝트는 어쨌든 그대로 진행하세요. 그러면 저희가 디펜스 중에 그 점을 고려해 보도록 하겠습니다. 그룹 리더가 없는 경우라 하더라도 여러분은 제출 디렉토리에 접근할 수 있습니다.
- 보너스 점수를 얻고 싶으면 다른 과제물도 제출해야 합니다.
- Moulinette는 용통성이 좀 없는 친구입니다. Norm을 따르지 않는 코드는 이해하려고 노력조차 하지 않을 겁니다. Moulinette는 Norminette라는 프로그램으로 파일이 norm을 따랐는지 확인합니다. 그러니까 Norminette의 확인을 통과하지 못하는 과제물을 제출한다는 건 어리석은 일이겠죠?



보너스를 고려해보기전에 팀에 부여된 과제가 완벽하게 작동하는지 꼭 확인해야합니다. 보너스과제가 작동하나 팀에 부여된 과제가 작동하지 않는다면 0 점입니다.



Norminette는 `-R CheckForbiddenSourceHeader` 플래그를 사용해 실행되어야 합니다. Moulinette도 이 플래그를 사용합니다.

# Chapter II

## Foreword

다음은 유명 TV 프로그램에 나오는 노래의 가사입니다.

[1절]

난 최고가 될 거야  
이전까지 아무도 없었던  
포켓몬을 잡는 건 내 진정한 시험  
포켓몬을 키우는 건 내 사명

난 대륙을 넘어다니며 여행을 할거야  
더 멀리, 더 넓게 찾으면서  
가르치고, 이해하면서  
포켓몬 내면의 힘을 말이지

[코러스]

포켓몬! 몽땅 잡고 말겠어! 너랑 내가 이건 내 운명이니까  
포켓몬! 그래, 너는 내 최고의 친구야 우리가 지켜야 될 이 세계에서 포켓몬! 진실한 마음으로  
용기는 우리를 고양시키지  
너는 나를 가르치고, 나는 너를 가르치고  
포켓몬! 몽땅 다 잡고 말겠어

[코러스]

우리가 가는 길에 있는 모든 도전  
용기를 가지고 마주할거야  
나는 매일 싸울 거야  
내 진짜 위치를 찾기 위해서  
나랑 가자, 시간이 됐어  
더 이상 좋은 팀은 없어  
손에 손잡고, 우리는 싸워 이길 거야  
우리의 꿈이 이뤄질 때까지!

[코러스]

분명 지금 노래를 따라 부르고 계시겠지만, 지금은 뭐 그려든 아니든 상관이 없습니다.  
참, 그리고 이 주제는 포켓몬스터와는 관련이 없습니다.

# Chapter III

## Main subject

|                     |   |
|---------------------|---|
|                     | Exercise 00                                 |
|                     | Rush0X                                      |
| Turn-in directory : | <i>ex00/</i>                                |
| Files to turn in :  | <code>main.c, ft_putchar.c, rush0X.c</code> |
| Allowed functions : | <code>write</code>                          |

- 제출할 파일: `main.c, ft_putchar.c`, 그리고 `rush0X.c`를 제출해야 합니다. 여기서 '`0X`'는 `rush` 번호입니다. 예: `rush00.c`.
- `main.c` 예:

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

- 따라서 `int`형 변수 2개(각각 `x, y`)를 인자로 받는 `rush` 함수를 반드시 작성해야 합니다.
- 여러분이 만든 `rush` 함수는 화면에 사각형을 표시해야 합니다. 이때 `x`는 사각형의 가로, `y`는 세로를 나타냅니다.
- 코렉션중에 여러분이 모든 요소를 올바르게 작성하였는지를 확인하기 위하여 그에 따라 `main`이 수정될 것입니다. 예를 들어 저희는 아래와 같은 테스트를 수행하여 여러분의 코드를 확인합니다.

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

# Chapter IV

## Rush 00

- rush(5,3)는 다음을 표시해야 합니다.:

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

- rush(5, 1)는 다음을 표시해야 합니다.:

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

- rush(1, 1)는 다음을 표시해야 합니다.:

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

- rush(1, 5)는 다음을 표시해야 합니다.:

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

- rush(4, 4)는 다음을 표시해야 합니다.:

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

# Chapter V

## Rush 01

- rush(5, 3)는 다음을 표시해야 합니다.:

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

- rush(5, 1)는 다음을 표시해야 합니다.:

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

- rush(1, 1)는 다음을 표시해야 합니다.:

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

- rush(1, 5)는 다음을 표시해야 합니다.:

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

- rush(4, 4)는 다음을 표시해야 합니다.:

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

# Chapter VI

## Rush 02

- rush(5,3)는 다음을 표시해야 합니다.:

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

- rush(5, 1)는 다음을 표시해야 합니다.:

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

- rush(1, 1)는 다음을 표시해야 합니다.:

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

- rush(1, 5)는 다음을 표시해야 합니다.:

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

- rush(4, 4)는 다음을 표시해야 합니다.:

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

# Chapter VII

## Rush 03

- rush(5,3)는 다음을 표시해야 합니다.:

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

- rush(5, 1)는 다음을 표시해야 합니다.:

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

- rush(1, 1)는 다음을 표시해야 합니다.:

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

- rush(1, 5)는 다음을 표시해야 합니다.:

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

- rush(4, 4)는 다음을 표시해야 합니다.:

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

# Chapter VIII

## Rush 04

- rush(5,3) should display :

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

- 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
$>
```