



C Piscine

C 11

*Summary:* 이 문서는 42 C Piscine 과정 C 11 모듈의 학습 주제입니다.

# Contents

I	Instructions	2
II	Foreword	4
III	Exercise 00 : ft_foreach	6
IV	Exercise 01 : ft_map	7
V	Exercise 02 : ft_any	8
VI	Exercise 03 : ft_count_if	9
VII	Exercise 04 : ft_is_sort	10
VIII	Exercise 05 : do-op	11
IX	Exercise 06 : ft_sort_string_tab	13
X	Exercise 07 : ft_advanced_sort_string_tab	14

# Chapter I

## Instructions

- 오직 이 페이지만 참고해야 합니다. 소문은 믿지 마세요.
- 파일 제출 전에 이 문서가 변경될 수도 있으니 주의하세요!
- 파일과 디렉토리에 대해 적절한 권한을 갖고 있는지 확인하세요.
- 모든 과제물을 제출할 때는 제출 절차를 따라야 합니다.
- 제출하신 과제물은 동료들끼리 서로 확인하고 평가하게 됩니다.
- 추가로, Moulinette라는 프로그램도 과제물을 확인하고 평가합니다.
- Moulinette는 아주 꼼꼼하고 깐깐하게 과제물을 평가합니다. 완전히 자동화된 프로그램이기 >때문에 일체의 협상은 불가능합니다. 그러니 좋지 않은 평가를 받고 실망하고 싶지 않다면 최선을 다해 철저하게 과제를 수행하세요.
- Moulinette는 그다지 마음이 너그럽지 못하답니다. 표준을 따르지 않는 코드는 이해하려고 노력조차 하지 않을 겁니다. Moulinette는 norminette라는 프로그램으로 파일이 표준을 따랐는지 확>인합니다. 그러니까 Norminette의 확인을 통과하지 못하는 과제물을 제출한다는 건 어리석은 일이>겠죠?
- Exercise는 난이도에 따라 쉬운 문제에서 어려운 문제 순으로 짜여 있습니다. 앞 단계 문제의 과제물이 완벽하게 작동하지 않으면 난이도가 더 높은 문제는 아무리 잘 완료했다 하더라도 평가에 반영되>지 않습니다.
- 사용이 금지된 함수를 사용하는 것은 부정 행위로 간주됩니다. 부정 행위는 -42점을 받게 되며, 받은 점수는 절대 조정이 불가능합니다.
- 프로그램을 제출해야 하는 연습 문제의 경우 main() 함수만 제출하면 됩니다.
- Moulinette는 -Wall -Wextra -Werror 플래그를 지정하여 컴파일하며 gcc를 사용합니다.
- 프로그램이 컴파일되지 않으면 0점을 받게 됩니다.
- 각 Exercise에서 정한 파일 이외의 어떠한 파일도 디렉토리에 남겨 두어서는 안 됩니다.
- 질문이 있으신가요? 오른쪽 동료에게 물어보세요. 아니면 왼쪽 동료에게 물어보세요.
- 참고 가이드는 Google / man / the Internet / ...입니다.

- 인트라넷의 포럼에서 ‘C Piscine’ 파트를 참조하거나 Slack의 Piscine 채널을 확인해 보세요.
- 예시를 꼼꼼히 살펴보세요. Exercise에서 명시적으로 언급되지 않은 세부적인 사항에 대한 힌트를 얻을 수도 있습니다.
- 오딘의 힘으로, 토르의 힘으로! 열심히 고민해 보세요!!!

# Chapter II

## Foreword

재밌는 얘기 하나 해드리죠.

(1982년 미국 캘리포니아) 로스앤젤레스 출신의 래리 월터스는 몇 안되는 다원상 후보 중 한 명으로, 운 좋게 살아 남아 자신의 사연을 들려줄 수 있게 되었다. "20년 동안 간절히 바래 온 꿈을 이뤘습니다." TV 광고 제작사에서 트럭 운전수로 일했던 그는 말했다.

"저는 아직 땅 위에 서 있어요. 그게 작동을 한다는 걸 입증했죠."

어릴 적 래리의 꿈은 나는 것이었다. 하지만 야속한 운명은 그가 꿈을 이루도록 두지 않았다. 공군에 들어갔지만 시력이 나빠 조종사가 될 수 없었다. 제대 후에 그는 뒤틀에 앉아 머리 위로 지나가는 비행기를 올려다볼 뿐이었다.

여느 날과 다름없이 '말도 못하게 편한' 시어스 접이식 의자에 앉아 시간을 보내던 그. 바로 그 의자에 앉아서였다. 그가 기상 관측용 풍선을 활용할 방법에 대한 아이디어를 떠올린 건. 그는 당장 육해군 군용 물품 상점으로 달려가 기상 관측용 풍선을 45개 사다가 접이식 의자에 끈으로 묶은 다음(영감 1이라고 이름을 붙였다) 직경 1.2 미터의 풍선을 헬륨 가스로 채웠다. 그리고는 샌드위치 몇 개, 밀리 라이트 맥주, 공기총 하나를 챙긴 다음 자기 몸을 의자에 끈으로 묶었다. 하강할 때가 되면 풍선 중에 몇 개를 공기총으로 쏴서 터뜨릴 생각이었다.

그의 계획은 묶어 놨던 닻을 자르면 뒤틀에서 한 9미터 정도 높이까지 천천히 떠오르는 것 이었고, 그 정도 높이에서 몇 시간쯤 유유자적하다 땅으로 내려오면 되겠다고 생각했다. 하지만 일은 그가 생각한대로 풀리지 않았다.

친구들이 그가 앉은 의자를 그의 지프차에 묶어 놨던 줄을 끊자, 그는 9미터까지 둑실 둑실 떠오르는 대신 마치 대포알처럼 LA 상공을 가르며 날아갔다. 각각 거의 1입방미터의 헬륨 가스가 들어 있는 45개의 헬륨 풍선에 실려서.

그렇게 솟구친 그를 실은 의자는 상공 30미터에서도, 300미터에서도 멈추지 않았다. 한참을 그렇게 계속 오르더니 거의 500미터 가까이 되어서야 멈췄다.

이 높이에 이르자 그는 풍선을 터뜨렸다는 까딱 잘못하면 균형을 잃고 진짜 위험한 상황에 처할 수 있겠다는 생각이 들었다. 결국 그 상태로 그는 맥주와 샌드위치를 움켜쥔 채 추위와 공포에 오들오들 떨며 14시간을 넘게 마냥 버텼다. 그가 LA 공항의 주요 접근 회랑을 지나가자, 트랜스월드항공과 델타항공의 조종사들은 어안이 벙벙해 이 기묘한 광경을 보고하는 무전 교신을 보내기도 했다.

결국 그는 간신히 용기를 내 풍선 몇 개를 쏘아 터뜨렸고 서서히 하강했다. 풍선을 묶고 있던 밧줄이 처져 매달려 있다가 근처 전선에 걸렸고 이 때문에 롱비치 지역이 20분간 정전되기도 했다. 그가 안전한 위치에 다다르자 그를 기다리고 있던 LA 경찰에게 체포되었다. 수갑을 찬 채 끌려가던 그에게 이 위험한 구출 작전을 취재하러 나온 기자가 왜 이런 짓을 했느냐고 물었다. 래리는 태연하게 말했다. "사람이 방구석에만 처박혀 지낼 수는 없잖아요."

미국 연방항공국으로서는 박수 쳐 줄 일은 아니었다. 안전 감독관 닐 세보이는 "그가 미 연방항공법의 일부 규정을 위반했다는 건 확실합니다. 어느 규정에 위반되는지에 대해 판단을 내리는 대로 벌금이 부과될 겁니다."라고 말했다.

이 이야기의 교훈은 래리 월터스가 의자에 가만히 앉아서 C나 배웠어야 했다는 겁니다.

# Chapter III

## Exercise 00 : ft\_foreach

	Exercise 00
	ft_FOREACH
Turn-in directory :	ex00/
Files to turn in :	ft_FOREACH.c
Allowed functions :	None

- 주어진 int형 배열에 대해 배열의 모든 요소에 함수를 적용하는 `ft_FOREACH`라는 함수를 작성하세요. 이 함수는 배열의 순서에 따라 적용되어야 합니다.
- 함수에 대해 프로토타입은 다음과 같이 선언합니다. :

```
void ft_FOREACH(int *tab, int length, void(*f)(int));
```

- 예를 들어 배열의 모든 int를 표시하기 위해 함수 `ft_FOREACH`를 다음과 같이 호출할 수 있습니다.

```
ft_FOREACH(tab, 1337, &ft_putstr);
```

# Chapter IV

## Exercise 01 : ft\_map

	Exercise 01
	ft_map
Turn-in directory :	ex01/
Files to turn in :	ft_map.c
Allowed functions :	malloc

- 주어진 int형 배열에 대해 배열의 모든 요소(순서대로)에 함수를 적용하고 모든 반환값의 배열을 반환하는 함수 `ft_map`을 작성하세요.
- 이 함수는 배열의 순서를 따라 적용되어야 합니다.
- 함수에 대해 프로토타입은 다음과 같이 선언합니다. :

```
int *ft_map(int *tab, int length, int(*f)(int));
```

# Chapter V

## Exercise 02 : ft\_any

	Exercise 02
	ft_any
Turn-in directory :	ex02/
Files to turn in :	ft_any.c
Allowed functions :	None

- 함수 f에 전달될 때 배열의 요소 중 최소한 하나가 0이외의 값을 반환할 경우 1을 반환하는 함수 ft\_any를 작성하세요. 그렇지 않은 경우에는 0을 반환해야 합니다.
- 이 함수는 배열의 순서를 따라 적용됩니다.
- 함수에 대해 프로토타입은 다음과 같이 선언합니다. :

```
int ft_any(char **tab, int(*f)(char*));
```

- 배열의 끝은 NULL포인터로 표시되어 있습니다.

# Chapter VI

## Exercise 03 : ft\_count\_if

	Exercise 03
	ft_count_if
Turn-in directory :	ex03/
Files to turn in :	ft_count_if.c
Allowed functions :	None

- 함수 f에 전달될 때 반환값으로 0을 반환하지 않는 배열의 요소의 개수를 반환하는 함수 `ft_count_if`를 작성하세요.
- 이 함수는 배열의 순서를 따라 적용되어야 합니다.
- 함수에 대해 프로토타입은 다음과 같이 선언합니다. :

```
int          ft_count_if(char **tab, int length, int (*f)(char*));
```

# Chapter VII

## Exercise 04 : ft\_is\_sort

	Exercise 04
	ft_is_sort
Turn-in directory :	ex04/
Files to turn in :	ft_is_sort.c
Allowed functions :	None

- 배열이 정렬되어 있으면 1을, 정렬되어 있지 않으면 0을 반환하는 함수 `ft_is_sort`를 작성하세요.
- 인자로 주어진 함수는 첫 번째 인자가 두 번째 인자보다 작을 경우 음의 정수를 반환하고, 같으면 0을 반환하고, 그 밖의 경우에는 양의 정수를 반환해야 합니다.
- 함수에 대해 프로토타입은 다음과 같이 선언합니다. :

```
int           ft_is_sort(int *tab, int length, int(*f)(int, int));
```

# Chapter VIII

## Exercise 05 : do-op

	Exercise 05
	do-op
Turn-in directory :	ex05/
Files to turn in :	Your program files
Allowed functions :	write

- do-op이라는 프로그램을 작성하세요.
- 이 프로그램은 다음의 세 개의 인자로 실행됩니다. do-op value1 operator value2
- 예:

```
$> ./do-op 42 "+" 21  
63  
$>
```

- 연산자를 처리하기 위해 함수를 가리키는 포인터의 배열을 사용해야 합니다.
- 유효하지 않은 연산자일 경우 프로그램은 0을 출력해야 합니다.
- 인자의 개수가 유효하지 않을 경우 do-op은 아무것도 표시하지 않습니다.
- 다음과 같은 연산자에 대해서는 프로그램이 그 결과를 받아 출력해야 합니다. '+', '-' , '/', '\*' , '^'
- 0으로 나누는 연산은 다음을 출력해야 합니다. :

```
Stop : division by zero
```

- 0으로 나눈 나머지 연산(modulo)의 경우 다음을 출력해야 합니다. :

```
Stop : modulo by zero
```

- 다음은 Moulinette가 실행할 테스트의 예입니다. :

```
$> make clean
$> make
$> ./do-op
$> ./do-op 1 + 1
2
$> ./do-op 42amis - ---20toto12
62
$> ./do-op 1 p 1
0
$> ./do-op 1 + toto3
1
$>
$> ./do-op toto3 + 4
4
$> ./do-op foo plus bar
0
$> ./do-op 25 / 0
Stop : division by zero
$> ./do-op 25 % 0
Stop : modulo by zero
$>
```

# Chapter IX

## Exercise 06 : ft\_sort\_string\_tab

	Exercise 06
	ft_sort_string_tab
Turn-in directory :	ex06/
Files to turn in :	ft_sort_string_tab.c
Allowed functions :	None

- tab의 문자열을 아스키(ascii) 순서로 정렬하는 함수 `ft_sort_string_tab`을 작성하세요.
- tab은 NULL종료됩니다.
- 정렬은 배열의 포인터를 교환하여 실행됩니다.
- 프로토타입은 다음과 같이 선언합니다. :

```
void ft_sort_string_tab(char **tab);
```

# Chapter X

## Exercise 07 : ft\_advanced\_sort\_string\_tab

	Exercise 07
	ft_advanced_sort_string_tab
Turn-in directory :	ex07/
Files to turn in :	ft_advanced_sort_string_tab.c
Allowed functions :	None

- 인자로 주어진 함수의 반환값에 따라 정렬하는 함수 `ft_advanced_sort_string_tab`을 작성하세요.
- 정렬은 배열의 포인터를 교환함으로써 이루어집니다.
- `tab`의 마지막은 `NULL`입니다.
- 프로토타입은 다음과 같이 선언합니다. :

```
void ft_advanced_sort_string_tab(char **tab, int(*cmp)(char *, char *));
```



Calling `ft_advanced_sort_string_tab()` with `ft_strcmp`를 두번 째 인자로 하여 함수 `ft_advanced_sort_string_tab()`을 호출할 경우 `ft_sort_string_tab()`와 동일한 결과가 출력됩니다.