



C Piscine

C 07

Staff 42 pedago@42.fr

Summary: 이 문서에서는 42 C Piscine 과정의 모듈 C 07에서 학습하는 주제를 다룹니다.

Contents

I	Instructions	2
II	Foreword	4
III	Exercise 00 : ft_strdup	5
IV	Exercise 01 : ft_range	6
V	Exercise 02 : ft_ultimate_range	7
VI	Exercice 03 : ft_strjoin	8
VII	Exercise 04 : ft_convert_base	9
VIII	Exercise 05 : ft_split	10

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 확인합니다. 그러니까 Norminette의 확인을 통과하지 못하는 과제물을 제출한다는 건 어리석은 일이겠죠?
- 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 반영되지 않습니다.
- 사용이 금지된 함수를 사용하는 것은 부정 행위로 간주됩니다. 부정 행위는 -42-42 점을 받게 되며, 받은 점수는 절대 조정이 불가능합니다.
- 프로그램을 제출해야 하는 문제의 경우 main() 함수만 제출하면 됩니다.
- Moulinette은 -Wall -Wextra -Werror 플래그를 지정하여 컴파일하며 gcc를 사용합니다.
- 프로그램이 컴파일되지 않으면 0점을 받게 됩니다.
- 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!!!



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

Chapter II

Foreword

모티: 릭!

릭: 헉, 헉, 헉.....모티, 네 거시기에서 손 떼! 우리가 자유롭게 대화할 유일한 방법이라고. 주변을 둘러봐, 모티. 이 세계가 진짜 같아? 바보라도 알겠다. 세부적인 게 다 허접하다고. 봐라, 소시지 두 개 사이에 빵을 끼워서 먹고 있잖아.

모티: 전 모르겠어요, 릭. 예전에도 저렇게 먹는 걸 봤거든요.

릭: 저 할망구를 봐. 고양이를 줄에 산책시키고 있다고.

모티: 스펜서 부인은 원래 항상 저래요.

릭: 봐, 난 스펜서 부인 얘기는 듣기도 싫다, 모티! 바보 같은 할망구라고! 그래, 그럼 저기를 봐. 저건 어때, 모티?

모티: 저도 저건 좀 이상하네요.

릭: 맙소사, 이제서야, 모티? 저건 현실에서 본 적 없는 게 확실하냐?

모티: 네, 저런 건 본 적이 없어요. 제 말은 식빵이 왜 토스터 안에 살겠어요? 식빵한테는 제일 무서운 장소일 텐데요. 무슨 말인지 아시겠죠?

릭: 그게 초점이 아니잖아, 모티. 걔가 왜 바퀴 달린 토스터를 운전하겠어? 너희 자동차는 너희 집 축소판처럼 생겼어? 아니지.

모티: 그래서 우린 어쩌죠? 그들이 원하는 게 뭘까요?

릭: 뻔하지, 모티. 네가 진작 잘 들었다면 알 텐데. [구급차가 릭과 모티를 지나쳐 멈춰선다. 뒷문이 열린다.]

구급대원: 미국 대통령이 타고 있습니다. 10cc 농축 암흑 물질이 필요해요. 아니면 대통령이 죽습니다!

모티: 농축 암흑 물질요? 수업에서도 그걸 물어봤어요.

릭: 그래, 그건 내가 발명한 특별한 연료거든. 누구보다 우주를 빠르게 여행할 수 있는 연료. 자이지리안은 언제나 내 비밀을 캐내려 일을 꾸민다니까. 하지만 이번엔 큰 실수한 거다. 넌 여기 끌어들이다니. 값을 톡톡히 치러야 할 거다.

모티: 어쩌시려고요? 아니, 이제 우린 어떻게 하죠?

릭: 사기꾼들에게 우리가 사기를 치는 거야, 모티. 그리고 걔네가 가진 모든 걸 뺏는 거지.

다음 연습 문제는 애니메이션 ‘릭 앤 모티’의 팬이라면 쉽게 풀 수 있을 겁니다.

Chapter III

Exercise 00 : ft_strdup

	Exercise 00
	ft_strdup
Turn-in directory :	<i>ex00/</i>
Files to turn in :	ft_strdup.c
Allowed functions :	malloc

- Reproduce the behavior of the function `strdup` (man `strdup`).
- 프로토타입은 다음과 같이 선언합니다.:

```
char *ft_strdup(char *src);
```

Chapter IV

Exercise 01 : ft_range

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

- int 배열을 반환하는 함수 ft_range를 작성하세요. 이 int 배열은 최소값(min)과 최대값(max) 사이의 모든 값을 포함해야 합니다.
- 최소값은 포함되며 최대값은 제외됩니다.
- 프로토타입은 다음과 같이 선언합니다.:

```
int *ft_range(int min, int max);
```

- 최소값 이 최대값보다 크거나 같으면 널(null) 포인터가 반환되어야 합니다.

Chapter V

Exercise 02 : ft_ultimate_range

	Exercise 02
	ft_ultimate_range
Turn-in directory :	ex02/
Files to turn in :	ft_ultimate_range.c
Allowed functions :	malloc

- int 배열을 할당하고 배치하는 함수 ft_ultimate_range를 작성하세요. 이 int 배열은 최소값(min)과 최대값(max) 사이의 모든 값을 포함해야 합니다.
- 최소값은 포함되며 최대값은 제외됩니다.
- 프로토타입은 다음과 같이 선언합니다.:

```
int ft_ultimate_range(int **range, int min, int max);
```

- 범위의 크기가 반환되어야 합니다. (오류 시에는 -1)
- 최소값이 최대값보다 크거나 같을 경우 범위에는 NULL 포인트가 발생해 0이 반환되어야 합니다.



Chapter VI

Exercice 03 : ft_strjoin

	Exercise 03
	ft_strjoin
Turn-in directory :	ex03/
Files to turn in :	ft_strjoin.c
Allowed functions :	malloc

- `strs`이 가리키는 모든 문자열을 `sep`로 구분하여 연결하는 함수를 작성하세요.
- `size`는 `strs` 내의 문자열의 개수입니다.
- `size`가 0이면, `free` 할 수 있는 빈 문자열이어야합니다.
- 프로토타입은 다음과 같이 선언합니다.:

```
char *ft_strjoin(int size, char **strs, char *sep);
```

Chapter VII

Exercise 04 : ft_convert_base

	Exercise 04
	ft_convert_base
Turn-in directory :	ex04/
Files to turn in :	ft_convert_base.c, ft_convert_base2.c
Allowed functions :	malloc, free

- 문자열 nbr을 base_from 진법에서 base_to 진법으로 전환한 결과를 반환하는 함수를 작성하세요.
- nbr, base_from, base_to는 수정하면 안될듯..
- nbr 은 ft_atoi_base (다른 모듈에서 학습한 내용)에서와 동일한 규칙을 따릅니다. '+', '-' 및 화이트 스페이스에 유의하세요.
- nbr이 표현하는 수는 int 안에 들어가야 합니다.
- 진법이 잘못된 경우, NULL이 반환됩니다.
- 반환되는 수의 앞에는 필요하면 '-' 한 개만 붙을 수 있습니다. 화이트 스페이스나 '+'는 붙일 수 없습니다.
- 프로토타입은 다음과 같이 선언합니다.:

```
char *ft_convert_base(char *nbr, char *base_from, char *base_to);
```

Chapter VIII

Exercise 05 : ft_split

	Exercise 05
	ft_split
Turn-in directory :	ex05/
Files to turn in :	ft_split.c
Allowed functions :	malloc

- 문자열을 다른 문자열을 기준으로 나누는 함수를 작성하세요.
- 문자열 charset의 각 문자는 str 문자열을 구분하기 위한 구분자로 사용해야 합니다.
- 이 함수는 배열의 각 항에 두 개의 구분자로 둘러싸인 문자열의 주소가 포함되어 있는 배열을 반환합니다. 배열의 마지막에는 0에 해당하는 값을 넣어 배열의 끝임을 표시해야 합니다.
- 배열에 빈 문자열은 없습니다. 이를 감안해 결론을 도출해 보세요.
- 인자로 주어진 문자열은 수정할 수 없습니다.
- 프로토타입은 다음과 같이 선언합니다.:

```
char **ft_split(char *str, char *charset);
```