



C Piscine

C 08

*Summary: Bu doküman C Piscine @ 42 içindeki C 08 modülünün dersidir .*

*Version:*

# Contents

I	Yönergeler	2
II	Önsöz	4
III	Egzersiz 00 : ft.h	5
IV	Egzersiz 01 : ft_boolean.h	6
V	Egzersiz 02 : ft_abs.h	8
VI	Egzersiz 03 : ft_point.h	9
VII	Egzersiz 04 : ft_strs_to_tab	10
VIII	Çalışma 05 : ft_show_tab	12
IX	Submission and peer-evaluation	13

# Chapter I

## Yönergeler

- Lütfen sadece bu sayfayı referans alınız: söylentilere kulak asmayınız.
- Dikkat! Dokümanın gönderim öncesinde değişme ihtimali vardır.
- Lütfen dosyalarınız ve dizileriniz için gerekli yetkilere sahip olduğunuzdan emin olunuz.
- Bütün çalışmalarınız için gönderim talimatlarını takip ediniz.
- Çalışmalarınız sınıf arkadaşlarınız tarafından kontrol edilip notlandırılacaktır.
- Aynı zamanda, çalışmalarınız Moulinette adlı program tarafından da kontrol edilip notlandırılacaktır.
- Moulinette değerlendirmelerinde çok titiz ve katıdır. Otomatik bir program olmasından dolayı görüş alışverişi mümkün değildir. Sürpriz bir sonuçla karşılaşmak için çalışmalarınızı dikkatlice yapınız.
- Moulinette çok açık görüşlü değildir. Kodunuz Norm'a uymadığı takdirde onu anlamaya çalışmayacaktır. Moulinette dosyalarınızın norm'a uyup uymadığını kontrol etmek için **norminette** adında bir program kullanmaktadır. TL;DR: **norminette**'in kontrolünden geçemeyecek bir dosya teslim etmek akılsızca olacaktır.
- Çalışmalar en kolaydan en zora olacak şekilde zorluklarına göre sıralanmıştır. Daha zor bir çalışma başarıyla tamamlanmış bile olsa daha kolay bir çalışmanın tamamıyla fonksiyonel olmaması durumunda dikkate **alınmayacaktır**.
- Yasaklanmış bir fonksiyon kullanmak hile olarak görülmektedir. Bunu yapan kişiler -42 puan alacaktır, ve bu not pazarlığa tabi değildir.
- Sizden program istersek sadece bir main() fonksiyonu göndermeniz gerekir.
- Moulinette çalışmaları şu şekilde sınıflandırır: -Wall -Wextra -Werror ve gcc
- Eğer programınız sınıflandırılmazsa, 0 alırsınız.
- Dizinizde konunun başlığındakiler dışında hiçbir dosya bırakmayınız.
- Bir sorunuz mu var? Sağınızdaki arkadaşınıza sorun. Olmadı solunuzdakine...

- Başvuru kılavuzunuzun adı `Google / man / the Internet / ...` 'dır.
- Intranetteki forumun "C Piscine" kısmını ya da Slack'deki Piscine bölümünü kontrol edin.
- Konu içerisinde net bir şekilde belirtilmemiş detayları anlayabilmek için örnekleri dikkatlice inceleyiniz.
- Odin ve Thor adına ! Kafayı çalıştırın !!!

# Chapter II

## Önsöz

Vikipedi 'de Ornitorenk hakkında şu bilgiler vardır:

Ornitorenk (*Ornithorhynchus anatinus*), ya da ördek-gagalı platipus, Tazmanya da dahil olmak üzere doğu Avustralya'ya endemik olan yarı sucul yumurtlayan Dört ekidne türüyle birlikte hala var olan beş monotrem, doğurmak yerine yumurtlayan Fakat, fosil kalıntılarında onunla ilişkili türler bulunmuştur.


Bu ördek-gagalı, kunduz kuyruklu, yumurtlayan memelinin olağandışı görünüşü Avrupalı

20. yüzyılın ilk başlarına kadar kürkü için avlanmıştır. Fakat şu anda bulunduğu alan

Bu ders tabi ki de ornitorenkten bahsetmemektedir.

# Chapter III

## Egzersiz 00 : ft.h


	Exercise 00
ft.h	
Turn-in directory : <i>ex00/</i>	
Files to turn in : <b>ft.h</b>	
Allowed functions : None	

- Kendi **ft.h** dosyanızı oluşturunuz.
- Aşağıdaki fonksiyonların prototiplerinin hepsini barındırmalıdır :

```
void    ft_putchar(char c);
void    ft_swap(int *a, int *b);
void    ft_putstr(char *str);
int     ft_strlen(char *str);
int     ft_strcmp(char *s1, char *s2);
```

# Chapter IV

## Egzersiz 01 : ft\_boolean.h

	Exercise 01
ft_boolean.h	
Turn-in directory : <i>ex01/</i>	
Files to turn in : <b>ft_boolean.h</b>	
Allowed functions : None	

- **ft\_boolean.h** dosyası oluşturunuz. Aşağıdaki main'i uygun bir şekilde sınıflandırıp çalıştırmalıdır:

```
#include "ft_boolean.h"

void      ft_putstr(char *str)
{
    while (*str)
        write(1, str++, 1);
}

t_bool     ft_is_even(int nbr)
{
    return ((EVEN(nbr)) ? TRUE : FALSE);
}

int        main(int argc, char **argv)
{
    (void)argv;
    if (ft_is_even(argc - 1) == TRUE)
        ft_putstr(EVEN_MSG);
    else
        ft_putstr(ODD_MSG);
    return (SUCCESS);
}
```

- Bu program şunu göstermelidir

```
I have an even number of arguments.
```

- ya da şunu göstermelidir

```
I have an odd number of arguments.
```

- Uygun olduğunda da bir satır sonu ile izlenmelidir.




Norminette `-R CheckDefine` işareti ile başlatılmalıdır . Moulinette de bunu kullanacaktır.



# Chapter V

## Egzersiz 02 : ft\_abs.h

	Exercise 02
	ft_abs.h
	Turn-in directory : <i>ex02/</i>
	Files to turn in : <b>ft_abs.h</b>
	Allowed functions : None

- Değişkenini onun mutlak değeriyle değiştiren bir makro ABS oluşturunuz:


```
#define ABS(Value)
```



Norminette *-R CheckDefine* işareti ile başlatılmalıdır . Moulinette de bunu kullanacaktır.

# Chapter VI

## Egzersiz 03 : ft\_point.h

	Exercise 03
ft_point.h	
Turn-in directory : <i>ex03/</i>	
Files to turn in : <b>ft_point.h</b>	
Allowed functions : None	

- Aşağıdaki main ile derlenebilen bir **ft\_point.h** dosyası oluşturunuz:

```
#include "ft_point.h"


void      set_point(t_point *point)
{
    point->x = 42;
    point->y = 21;
}

int       main(void)
{
    t_point    point;

    set_point(&point);
    return (0);
}
```

# Chapter VII

## Egzersiz 04 : ft\_strs\_to\_tab

	Exercise 04
ft_strs_to_tab	
Turn-in directory : <i>ex04/</i>	
Files to turn in : <b>ft_strs_to_tab.c</b>	
Allowed functions : <b>malloc, free</b>	

- Bir matrisin dizilerini değişken olarak alan ve bu matrisin boyutunu alan bir fonksiyon oluşturunuz.
- Prototip şu şekilde olmalıdır:

```
struct s_stock_str *ft_strs_to_tab(int ac, char **av);
```

- av'nin her ögesini bir yapıya dönüştürecektir.
- Bu yapı bizim tarafımızdan temin edilen **ft\_stock\_str.h** dosyasında şu şekilde tanımlanacaktır:


```
typedef struct s_stock_str
{
    int size;
    char *str;
    char *copy;
} t_stock_str;
```

- size dizinin uzunluğu olacak şekilde;
  - str dizi olacak şekilde;
  - copy dizinin bir kopyası olacak şekilde;
- av'nin sırasını tutmalıdır.

- Geri dödürülen matrise bellekte yer açılmalı ve son ögesinin `str` 'si sıfır yapılmalıdır. Bu matrisin sonuna işaret edecektir.
- Bir hata olursa NULL işaretçi döndürülmelidir
- Fonksiyonunuzu, kendi `ft_show_tab` (bir sonraki çalışma) fonksiyonumuzla test edeceğiz. Bunu göz önüne alarak başarmaya çalışın !

# Chapter VIII

## Çalışma 05 : ft\_show\_tab

	Exercise 05
ft_show_tab	
Turn-in directory : <i>ex05/</i>	
Files to turn in : <b>ft_show_tab.c</b>	
Allowed functions : <b>write</b>	

- Bir önceki fonksiyon tarafından yaratılan matrisin içeriğini gösteren bir fonksiyon oluşturunuz.
- Prototip şu şekilde olmalıdır:

```
void ft_show_tab(struct s_stock_str *par);
```

- Yapı bir önceki egzersizdeki ile aynı olacaktır ve **ft\_stock\_str.h** dosyasında tanımlanacaktır.
- Her öge için aşağıdakileri göstereceğiz:
  - Dizin ardından '\n' gelecek
  - Boyutun ardından '\n' gelecek
  - Dizin kopyasının (düzenlenmiş olabilir) ardından '\n' gelecek
- Fonksiyonunuzu, kendi **ft\_show\_tab** (bir sonraki çalışma) fonksiyonumuzla test edeceğiz. Bunu göz önüne alarak başarmaya çalışın !

# Chapter IX

## Submission and peer-evaluation

Egzersizlerinizi yollarken Git deponuzu kullanıcaksınız her zamanki gibi. Savunmada sadece deponuzun içindekiler değerlendirilicektir. Dosya ve klasör isimlerini bir daha kontrol etmekten çekinmeyin. Doğru olduklarına emin olun.



Yalnızca talep edilen dosyaları teslim etmeniz gerekir.



