



C Piscine

C 00

Summary: Bu doküman C Piscine @ 42 içindeki C 00 modülünün konusudur.

Contents

I	Yönergeler	2
II	Önsöz	4
III	Çalışma 00 : ft_putchar	5
IV	Çalışma 01 : ft_print_alphabet	6
V	Çalışma 02 : ft_print_reverse_alphabet	7
VI	Çalışma 03 : ft_print_numbers	8
VII	Çalışma 04 : ft_is_negative	9
VIII	Çalışma 05 : ft_print_comb	10
IX	Çalışma 06 : ft_print_comb2	11
X	Çalışma 07 : ft_putnbr	12
XI	Çalışma 08 : ft_print_combn	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. Süpriz 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 !!!



Norminette `-R CheckForbiddenSourceHeader` bayrağı ile başlatılmalıdır
. Moulinette de bunu kullanacaktır.

Chapter II

Önsöz

Morina balığı ciğeri yağı, Morina balığının ciğerinden elde edilen takviye edici bir

Çoğu balık yağında olduğu gibi, bunda da yüksek miktarda omega-3 yağ asitleri, eikosa ve dokosaheksaenoik asit (DHA) bulunmaktadır.

Morina balığı ciğeri yağı aynı zamanda A vitamini ve D vitamini içermektedir.


Tarih boyunca içindeki A vitamini ve D vitamini için kullanılmıştır.

İçindeki D vitaminin raşitizm ve D vitamini eksikliğinin diğer semptomlarını önlemesi zamanında Morina balığı ciğeri yağı sıklıkla çocuklara verilmekteydi.

Morina balığı ciğeri yağının yerine, C iyidir, biraz yiyin!

Chapter III

Çalışma 00 : ft_putchar

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

- Karakteri bir parametre olarak gösteren bir fonksiyon yazınız.
- Şu şekilde modellenektir :


```
void ft_putchar(char c);
```

Karakteri göstermek için, `write` fonksiyonunu aşağıdaki gibi kullanmalısınız.

```
write(1, &c, 1);
```

Chapter IV

Çalışma 01 : ft_print_alphabet

	Exercise 01
ft_print_alphabet	
Turn-in directory : <i>ex01/</i>	
Files to turn in : ft_print_alphabet.c	
Allowed functions : write	


- 'a' harfinden başlayarak, artan şekilde (a'dan z'ye doğru), tek satırda ve hepsi küçük harf olarak alfabeyi gösteren bir fonksiyon oluşturunuz.
- Şu şekilde modelleneyecektir :

```
void ft_print_alphabet(void);
```

Chapter V

Çalışma 02 :

ft_print_reverse_alphabet


	Exercise 02
ft_print_reverse_alphabet	
Turn-in directory : <i>ex02/</i>	
Files to turn in : ft_print_reverse_alphabet.c	
Allowed functions : write	

- 'z' harfinden başlayarak, azalan şekilde (z'den a ya doğru), tek satırda ve hepsi küçük harf olarak alfabeyi gösteren bir fonksiyon oluşturunuz.
- Şu şekilde modelleneyecektir :

```
void ft_print_reverse_alphabet(void);
```


Chapter VI

Çalışma 03 : ft_print_numbers


	Exercise 03
ft_print_numbers	
Turn-in directory : <i>ex03/</i>	
Files to turn in : ft_print_numbers.c	
Allowed functions : write	

- Bütün rakamları tek bir satırda, küçükten büyüğe artan şekilde gösteren bir fonksiyon oluşturunuz.
- Şu şekilde modelleneyecektir :

```
void ft_print_numbers(void);
```

Chapter VII

Çalışma 04 : ft_is_negative


	Exercise 04
ft_is_negative	
Turn-in directory : <i>ex04/</i>	
Files to turn in : ft_is_negative.c	
Allowed functions : write	

- Parametre olarak tam sayının işareti girildiğinde 'N' ya da 'P' gösteren bir fonksiyon oluşturunuz. Eğer *n* negatifse, fonksiyon 'N' göstermeli. Eğer *n* pozitif ya da sıfır ise , fonksiyon 'P' göstermeli.
- Şu şekilde modelleneyecektir :

```
void ft_is_negative(int n);
```

Chapter VIII

Çalışma 05 : ft_print_comb

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

- Rakamları küçükten büyüğe doğru sıralanmış şekilde, üç farklı rakamın bütün farklı kombinasyonlarını artan şekilde gösteren bir fonksiyon oluşturunuz. - evet, tekrar etmek gönüllü.
- İstenilen çıktı aşağıdaki gibidir :


```
$>./a.out | cat -e
012, 013, 014, 015, 016, 017, 018, 019, 023, ..., 789$>
```

- 987 burada yok çünkü 789 hali hazırda bulunmakta.
- 999 burada yok çünkü 9 rakamı birden fazla kere bulunmakta.
- Şu şekilde modellenmelidir :

```
void ft_print_comb(void);
```

Chapter IX

Çalışma 06 : ft_print_comb2

	Exercise 06
ft_print_comb2	
Turn-in directory : <i>ex06/</i>	
Files to turn in : ft_print_comb2.c	
Allowed functions : write	

- 00 ve 99 aralığında iki rakamdan oluşan sayıların olası tüm kombinasyonlarını artan sırada gösteren bir fonksiyon oluşturunuz.
- Beklenen çıktı aşağıdaki gibidir :


```
$>./a.out | cat -e
00 01, 00 02, 00 03, 00 04, 00 05, ..., 00 99, 01 02, ..., 97 99, 98 99$>
```

- Şu şekilde modellenmelidir :

```
void ft_print_comb2(void);
```

Chapter X

Çalışma 07 : ft_putnbr

	Exercise 07
ft_putnbr	
Turn-in directory : <i>ex07/</i>	
Files to turn in : ft_putnbr.c	
Allowed functions : write	


- Girilen sayıyı bir parametre olarak gösteren bir fonksiyon oluşturunuz. Fonksiyon, `int` tarzı değişken içindeki olası bütün değerleri gösterebilir olmalıdır.
- Şu şekilde modellenmelidir :

```
void ft_putnbr(int nb);
```

- Örneğin:
 - `ft_putnbr(42)` , "42" 'i gösterir.

Chapter XI

Çalışma 08 : ft_print_combn

	Exercise 08
ft_print_combn	
Turn-in directory : <i>ex08/</i>	
Files to turn in : ft_print_combn.c	
Allowed functions : write	

- **n** miktarda sayının oluşturabileceği olası bütün kombinasyonları artan şekilde gösteren bir fonksiyon oluşturunuz.
- **n** şu şekilde olacaktır : $0 < n < 10$.
- Eğer **n** = 2 ise, beklenen çıktı şu şekilde olacaktır :

```
$>./a.out | cat -e
01, 02, 03, ..., 09, 12, ..., 79, 89$>
```

- Şu şekilde modellenmelidir :

```
void ft_print_combn(int n);
```