

C Piscine C 02

 $\ddot{O}zet: \ \ Bu \ doküman \ C \ Piscine \ @ \ 42 \ içindeki \ C \ 02 \ modülünün \ konusudur \ .$ 

Versiyon: 5.2

# İçindekiler

1	Yönergeler	2
II	Önsöz	4
III	Çalışma 00 : ft_strcpy	5
IV	Egzersiz 01 : ft_strncpy	6
V	Egzersiz 02 : ft_str_is_alpha	7
VI	Egzersiz 03 : ft_str_is_numeric	8
VII	Egzersiz 04 : ft_str_is_lowercase	9
VIII	Egzersiz 05 : ft_str_is_uppercase	10
IX	Egzersiz 06 : ft_str_is_printable	11
$\mathbf{X}$	Egzersiz 07 : ft_strupcase	12
XI	Egzersiz 08 : ft_strlowcase	13
XII	Egzersiz 09 : ft_strcapitalize	14
XIII	Egzersiz 10 : ft_strlcpy	15
XIV	Egzersiz 11 : ft_putstr_non_printable	16
XV	Egzersiz 12 : ft_print_memory	17
XVI	Submission and peer-evaluation	19

#### Bölüm 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 karşılaşmamak 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 sınıflandırılamazsa, 0 alırsınız.
- Dizininizde konunun başlığındakiler dışında hiçbir dosya bırakmayınız.
- Bir sorunuz mu var? Sağınızdaki arkadaşınıza sorun. Olmadı solunuzdakine...

- $\bullet$  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 işareti ile başlatılmalıdır . Moulinette de bunu kullanacaktır.

#### Bölüm II

#### Önsöz

Aşağıda Silicon Valley dizisindeki bir tartışmadan alıntılar vardır :

- Niye Emacs yerine Vim kullanmıyorsun ki? (KIKIRDAR)
- Zaten Emac yerine Vim kullanıyorum.
- Ah, Tanrım, bizi kurtar! Tamam, biliyor musun? Bunun yürüyeceğini sanmıyorum artık. Özür dilerim. Yani, ne yapacağız, dünyaya kafalarında böyle şeyler olan çocuklar mı getireceğiz? Bu onlara da haksızlık olmaz mı sence de?
- Çocuklar mı? Daha beraber bile olmadık.
- Ve bil bakalım, şu andan sonra böyle bir şey asla olamaz, çünkü ben hayatta tab yerine boşluk kullanan biriyle olmam.
- Richard! (BOŞLUK TUŞUNA BİRÇOK KEZ BASAR)
- Vay. Peki. Hoşçakal.
- Tek bir tab sana 8 boşluk kazandırıyor! (KAPI ÇARPILIR) (GÜRÜLTÜLÜ)

. .

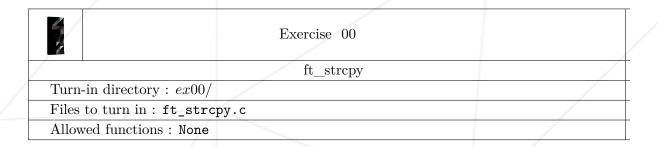
#### (RICHARD İNLER)

- Aman Tanrım! Richard, ne oldu?
- Merdivenlerden 8 basamağı aynı anda inmeye çalıştım. Bir şeyim yok ama.
- Görüşürüz, Richard.
- Sadece argümanımı kanıtlamaya çalışıyorum.

Umarız ki, aşağıdaki çalışmaları tamamlamak için emacs ve boşluk tuşunuzu kullanmak zorunda kalmazsınız.

### Bölüm III

# Çalışma 00 : ft\_strcpy

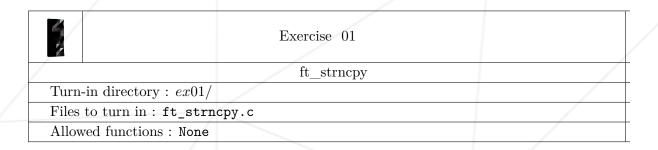


- strcpy fonksiyonun davranışını yeniden üret (man strcpy).
- Prototip şu şekilde olmalıdır :

char \*ft\_strcpy(char \*dest, char \*src);

## Bölüm IV

# Egzersiz 01: ft\_strncpy

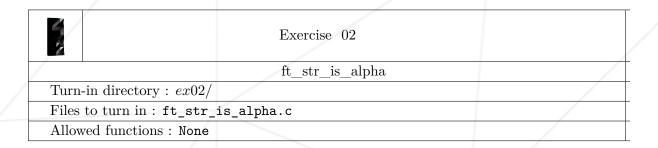


- strncpy fonksiyonun davranışını yeniden üret (man strncpy).
- Prototip şu şekilde olmalıdır :

char \*ft\_strncpy(char \*dest, char \*src, unsigned int n);

#### Bölüm V

# Egzersiz 02 : ft\_str\_is\_alpha

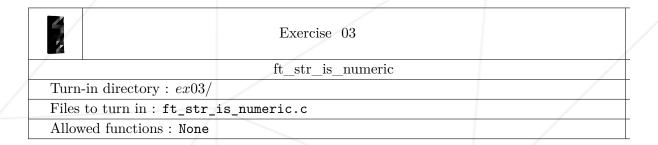


- Parametre olarak verilen dizin sadece alfabetik karakterler içeriyorsa 1, herhangi başka bir karakter içeriyorsa 0 sonucunu veren bir fonksiyon oluşturunuz.
- Prototip şu şekilde olmalıdır :

int ft\_str\_is\_alpha(char \*str);

#### Bölüm VI

Egzersiz 03 : ft\_str\_is\_numeric

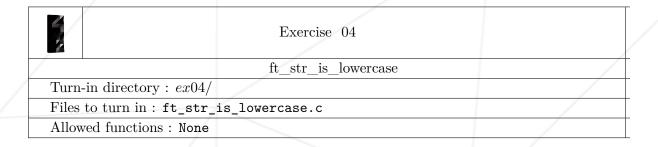


- Parametre olarak verilen dizin sadece rakam içeriyorsa 1, herhangi başka bir karakter içeriyorsa 0 sonucunu veren bir fonksiyon oluşturunuz.
- Prototip şu şekilde olmalıdır :

int ft\_str\_is\_numeric(char \*str);

#### Bölüm VII

Egzersiz 04 : ft\_str\_is\_lowercase

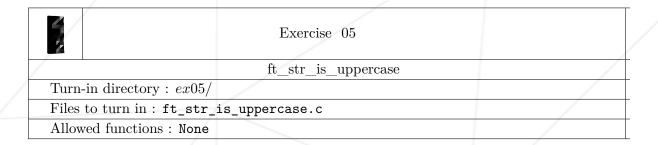


- Parametre olarak verilen dizin sadece küçük harfli alfabetik karakterler içeriyorsa 1, herhangi başka bir karakter içeriyorsa 0 sonucunu veren bir fonksiyon oluşturunuz.
- Prototip şu şekilde olmalıdır :

int ft\_str\_is\_lowercase(char \*str);

#### Bölüm VIII

Egzersiz  $05: ft\_str\_is\_uppercase$ 

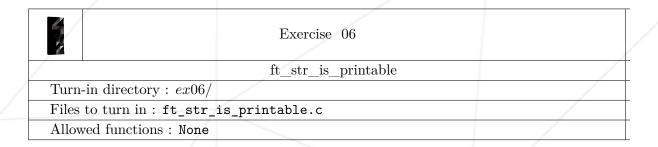


- Parametre olarak verilen dizin sadece büyük harfli alfabetik karakterler içeriyorsa 1, herhangi başka bir karakter içeriyorsa 0 sonucunu veren bir fonksiyon oluşturunuz.
- Prototip şu şekilde olmalıdır :

int ft\_str\_is\_uppercase(char \*str);

#### Bölüm IX

Egzersiz 06 : ft\_str\_is\_printable

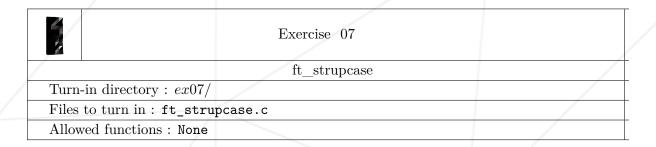


- Parametre olarak verilen dizin sadece yazdırılabilir karakterler içeriyorsa 1, herhangi başka bir karakter içeriyorsa 0 sonucunu veren bir fonksiyon oluşturunuz.
- Prototip şu şekilde olmalıdır :

int ft\_str\_is\_printable(char \*str);

#### Bölüm X

# Egzersiz 07: ft\_strupcase



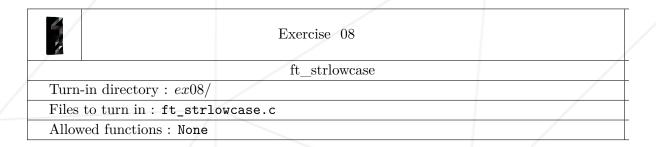
- Bütün harfleri büyük harfe dönüştüren bir fonksiyon oluşturunuz.
- Prototip şu şekilde olmalıdır :

char \*ft\_strupcase(char \*str);

• str sonucunu vermelidir.

#### Bölüm XI

# Egzersiz 08: ft\_strlowcase



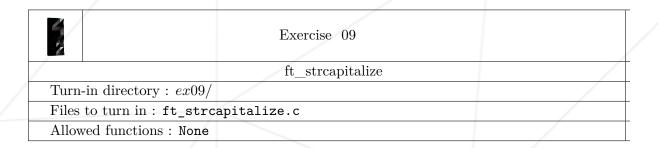
- Bütün harfleri küçük harfe dönüştüren bir fonksiyon oluşturunuz.
- Prototip şu şekilde olmalıdır :

char \*ft\_strlowcase(char \*str);

• str sonucunu vermelidir.

#### Bölüm XII

## Egzersiz 09: ft\_strcapitalize



- Bütün kelimelerin ilk harfini büyük harf yapan ve kalan harflerin hepsini küçük harfe yapan bir fonksiyon oluşturunuz.
- Kelime, alfanumerik karakterlerden oluşur.
- Prototip şu şekilde olmalıdır :

```
char *ft_strcapitalize(char *str);
```

- str sonucunu vermelidir.
- Örneğin:

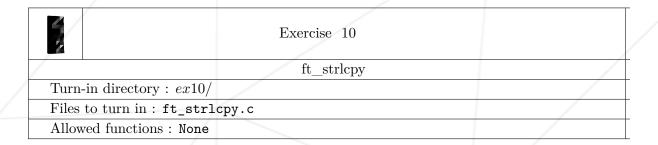
```
salut, comment tu vas ? 42mots quarante-deux; cinquante+et+un
```

• Şuna dönüşür:

Salut, Comment Tu Vas ? 42mots Quarante-Deux; Cinquante+Et+Un

#### Bölüm XIII

# Egzersiz 10: ft\_strlcpy

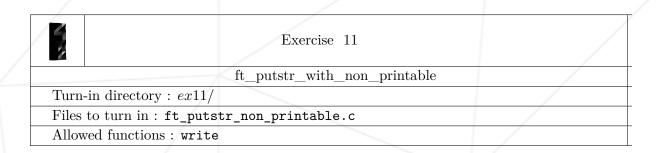


- strlcpy fonksiyonun davranışını yeniden üret (man strlcpy).
- Prototip şu şekilde olmalıdır :

unsigned int ft\_strlcpy(char \*dest, char \*src, unsigned int size);

#### Bölüm XIV

# Egzersiz 11: ft\_putstr\_non\_printable



- Karakter dizisini ekranda gösteren bir fonksiyon oluşturunuz. Eğer bu dizi yazdırılamaz karakter barındırıyorsa, bu karakterler, öncesinde "Ters eğik çizgi" gelecek biçimde onaltılık sayı sisteminde gösterilmelidir(küçük harf).
- Örneğin :

Coucou\ntu vas bien ?

• Fonksiyon şunu göstermelidir :

Coucou\Oatu vas bien ?

• Prototip şu şekilde olmalıdır :

roid ft\_putstr\_non\_printable(char \*str);

#### Bölüm XV

# Egzersiz 12: ft\_print\_memory

	Exercise 12	
	ft_print_memory	
Turn-in directory : $ex12$		
Files to turn in : ft_pr		
Allowed functions : wri		

- Belleği ekranda gösteren bir fonksiyon oluşturunuz.
- Bu bellek alanı görüntüsü birer boşluk ile birbirinden ayrılmış üç "sütun" a bölünmelidir :
  - o İlk satırın ilk karakterinin onaltılık adresini ':' izlemelidir.
  - o Onaltılıktaki içerik her iki karakterde bir boşluk vardır ve gerektiğinde boşluklarla takviye edilmelidir (aşağıdaki örneğe bakınız).
  - o İçerik yazdırılabilir karakterler şeklindedir.
- Eğer bir karakter yazdırılabilir değilse, bir nokta ile değiştirilir
- Her satırda onaltı karakter olmalıdır.
- Eğer size 0'a eşitse, hiçbir şey gösterilmemelidir.

C Piscine

• Örnek:

```
$> ./ft_print_memory
00000010a161f40: 426f 6e6a 6f75 7220 6c65 7320 616d 696e Bonjour les amin
000000010a161f50: 6368 6573 090a 0963 2020 6573 7420 666f ches...c est fo
000000010a161f60: 7509 746f 7574 0963 6520 7175 206f 6e20 u.tout.ce qu on
000000010a161f70: 7065 7574 2066 6169 7265 2061 7665 6309 peut faire avec.
000000010a161f80: 0a09 7072 696e 745f 6d65 6d6f 7279 0a0a ..print_memory..
000000010a161f90: 0a09 6c6f 6c2e 6c6f 6c0a 2000 ..lol.lol. .

$> ./ft_print_memory | cat -te
0000000107ff9f40: 426f 6e6a 6f75 7220 6c65 7320 616d 696e Bonjour les amin$
0000000107ff9f50: 6368 6573 090a 0963 2020 6573 7420 666f ches...c est fo$
0000000107ff9f60: 7509 746f 7574 0963 6520 7175 206f 6e20 u.tout.ce qu on $
0000000107ff9f70: 7065 7574 2066 6169 7265 2061 7665 6309 peut faire avec.$
0000000107ff9f80: 0a09 7072 696e 745f 6d65 6d6f 7279 0a0a ..print_memory..$
0000000107ff9f90: 0a09 6c6f 6c2e 6c6f 6c0a 2000 ..lol.lol. .$
$>
```

• Prototip şu şekilde olmalıdır :

```
void *ft_print_memory(void *addr, unsigned int size);
```

• addr sonucunu vermelidir.

#### Bölüm XVI

## 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.