



C Piscine

C 02

Özet: Bu doküman C Piscine @ 42 içindeki C 02 modülünün konusudur .

Versiyon: 5.3

İçindekiler

I	Yönergeler	2
II	Önsöz	4
III	Çalışma 00 : ft_strcpy	6
IV	Egzersiz 01 : ft_strncpy	7
V	Egzersiz 02 : ft_str_is_alpha	8
VI	Egzersiz 03 : ft_str_is_numeric	9
VII	Egzersiz 04 : ft_str_is_lowercase	10
VIII	Egzersiz 05 : ft_str_is_uppercase	11
IX	Egzersiz 06 : ft_str_is_printable	12
X	Egzersiz 07 : ft_strupcase	13
XI	Egzersiz 08 : ft_strlowercase	14
XII	Egzersiz 09 : ft_strcapitalize	15
XIII	Egzersiz 10 : ft_strlcpy	16
XIV	Egzersiz 11 : ft_putstr_non_printable	17
XV	Egzersiz 12 : ft_print_memory	18
XVI	Submission and peer-evaluation	20

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 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` 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 Emacs 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. Hoşcakal.
- 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.


Taban puanı

Bu projenin geiş taban puanı 50'dir.

Hangi egzersizin bu tabana ulaşmanızı sağlayacağını ve daha fazla egzersiz yapmak isteyip istemediğinizi belirlemek size kalmıştır.

Bölüm III

Çalışma 00 : ft_strcpy


	Exercise 00
ft_strcpy	
Turn-in directory : <i>ex00/</i>	
Files to turn in : ft_strcpy.c	
Allowed functions : None	

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

```
char      *ft_strcpy(char *dest, char *src);
```

Bölüm IV

Egzersiz 01 : ft_strncpy


	Exercise 01
ft_strncpy	
Turn-in directory : <i>ex01/</i>	
Files to turn in : ft_strncpy.c	
Allowed functions : None	

- **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

	Exercise 02
ft_str_is_alpha	
Turn-in directory : <i>ex02/</i>	
Files to turn in : ft_str_is_alpha.c	
Allowed functions : None	


- 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);
```

- Eğer **str** boşsa 1 sonucunu vermelidir.

Bölüm VI

Egzersiz 03 : ft_str_is_numeric

	Exercise 03
ft_str_is_numeric	
Turn-in directory : <i>ex03/</i>	
Files to turn in : ft_str_is_numeric.c	
Allowed functions : None	


- 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);
```

- Eğer **str** boşsa 1 sonucunu vermelidir.

Bölüm VII

Egzersiz 04 : ft_str_is_lowercase

	Exercise 04
ft_str_is_lowercase	
Turn-in directory : <i>ex04/</i>	
Files to turn in : ft_str_is_lowercase.c	
Allowed functions : None	


- 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);
```

- Eğer **str** boşsa 1 sonucunu vermelidir.

Bölüm VIII

Egzersiz 05 : ft_str_is_uppercase

	Exercise 05
ft_str_is_uppercase	
Turn-in directory : <i>ex05/</i>	
Files to turn in : ft_str_is_uppercase.c	
Allowed functions : None	


- 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);
```

- Eğer **str** boşsa 1 sonucunu vermelidir.

Bölüm IX

Egzersiz 06 : ft_str_is_printable

	Exercise 06
ft_str_is_printable	
Turn-in directory : <i>ex06/</i>	
Files to turn in : ft_str_is_printable.c	
Allowed functions : None	


- 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);
```

- Eğer **str** boşsa 1 sonucunu vermelidir.

Bölüm X

Egzersiz 07 : ft_strupcase

	Exercise 07
ft_strupcase	
Turn-in directory : <i>ex07/</i>	
Files to turn in : ft_strupcase.c	
Allowed functions : None	


- 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

	Exercise 08
ft_strlowcase	
Turn-in directory : <i>ex08/</i>	
Files to turn in : ft_strlowcase.c	
Allowed functions : None	


- 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

	Exercise 09
ft_strcapitalize	
Turn-in directory : <i>ex09/</i>	
Files to turn in : ft_strcapitalize.c	
Allowed functions : None	

- 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 dönüştür:

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


Bölüm XIII

Egzersiz 10 : ft_strlcpy


	Exercise 10
ft_strlcpy	
Turn-in directory : <i>ex10/</i>	
Files to turn in : ft_strlcpy.c	
Allowed functions : None	

- **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

	Exercise 11
ft_putstr_with_non_printable	
Turn-in directory : <i>ex11/</i>	
Files to turn in : ft_putstr_non_printable.c	
Allowed functions : write	

- Karakter dizisini ekranda gösteren bir fonksiyon oluşturunuz. Eğer bu dizi yazdırılmaz 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\0atu vas bien ?
```

- Prototip şu şekilde olmalıdır :

```
void      ft_putstr_non_printable(char *str);
```

Bölüm XV

Egzersiz 12 : ft_print_memory

	Exercise 12
ft_print_memory	
Turn-in directory : <i>ex12/</i>	
Files to turn in : ft_print_memory.c	
Allowed functions : write	

- Belleği ekranda gösteren bir fonksiyon oluşturunuz.
- Bu bellek alanı görüntüsü birer boşluk ile birbirinden ayrılmış üç "sütun" a bölünmelidir :
 - İlk satırın ilk karakterinin onaltılık adresini ':' izlemelidir.
 - Onaltılıktaki içerik her iki karakterde bir boşluk vardır ve gerektiğinde boşluklarla takviye edilmelidir (aşağıdaki örneğe bakınız).
 - 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.

- Örnek:

```
$> ./ft_print_memory
000000010a161f40: 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 kullanırsınız her zamanki gibi. Savunmada sadece deponuzun içindekiler değerlendirilecektir. Dosya ve klasör isimlerini bir daha kontrol etmekten çekinmeyin. Doğru olduklarına emin olun.



Yalnızca talep edilen dosyaları teslim etmeniz gerekir.