



C Piscine

Rush 01

Summary: Bu döküman, 42'deki C Piscine'sindeki Rush01'in dersidir.

Version: 7.2

Contents

I	Yönergeler	2
II	Önsöz	3
III	subject	4
IV	Annex	6
V	Submission and peer-evaluation	8

Chapter I

Yönergeler

- Grup, otomatik olarak savunma için kaydettirilecektir.
- İptal etmeyin. İkinci bir şansınız yok.
- Ders hakkında herhangi bir soru, dersi karmaşık hale getirecektir.
- Bütün çalışmalarınız için gönderim prosedürlerini takip etmek zorunludur.
- Bu ders, gönderimden bir saat öncesine kadar değişebilir.
- Moulinette gcc kullanır ve şu işaretleri kullanarak derler: -Wall -Wextra -Werror
- Programınız derlenmezse 0 alırsınız.
- Programınız Norm kurallarını göre yazılmalıdır. Bonus fonksiyon/dosyanız var ise norm kurallarına uyulmadığı takdirde de 0 verilecektir.
- Projeyi belirlenmiş ekibinizle yapıp, seçtiğiniz savunma vaktinde, bütün ekibinizle beraber orada olmalısınız.
- Savunma vakti geldiğinde, projeniz birmiş olmalıdır. Savunmanın amacı, çalışmanızın bütün detaylarını sumanız ve açıklamanızdır.
- Grubunuzun her üyesi, projenizin bütün detaylarından haberdar olmalıdır. Eğer iş bölümü yapmaya karar vererseniz, bütün üyelerin birbirlerinin ne yaptığından haberdar olduğundan emin olun. Savunma sırasında sorular cevaplandıracaksınız ve en kötü açıklama baz alınarak notlandırılacaksınız.
- Grubu bir araya getirmek sizin sorumluluğunuz. Grup arkadaşlarınızla iletişime geçmek için her türlü imkana sahipsiniz: telefon, elektronik posta, posta güvercini, spiritualizm vs. Yani, bahane üretmek için zaman harcamayın. Hayat her zaman adil değil, durum, ne yazık ki bu.
- Ancak, her şeyi denediyseniz ve hala bir arkadaşınıza ulaşamıyorsanız: projeyi yine de yapın ve savunma sırasında ne yapabileceğimize bakarız. Grup lideri bile kayıp olsa, gönderim dizinine erişim izniniz vardır.
- Keyfini çıkartın!

Chapter II

Önsöz


Rastgele birkaç filmden havalı alıntılar :

1. "Gerçekten özgün bir fikir bulmak. Kendimi başkalarından ayırabileceğim tek yol bu."
-Akıl Oyunları
2. "Kötü adam olmana gerek yok. Evrendeki en yetenekli, en ilginç ve en sıradışı insan."
-Lego Filmi
3. "Bazen, kimsenin hiçbir şey beklemeyeceği insanlar, kimsenin hayal edemeyeceği şeyler."
-Her Şeyin Teorisi
4. "İnsan gayretinin herhangi bir sınırı olmamalı. Hepimiz farklıyız. Hayat ne kadar büyük."
-X-Men Geçmiş Günler Gelecek
5. "Biri sadece tökezledi ve yolunu kaybetti diye sonsuza dek kayboldukları anlamına gelmez."
-Geleceğe Dönüş
6. "Gittiğimiz yerde yollara ihtiyacımız yok."
-Oyunbozan Ralph
7. "Kötüyüm ve bu, iyi. Hiç iyi olmayacağım, ve bu kötü değil. Kendimden başka olmak."
-Çeşitli filmler
8. "KA-ME-HA-ME-HAAAAAAAAAAAA"

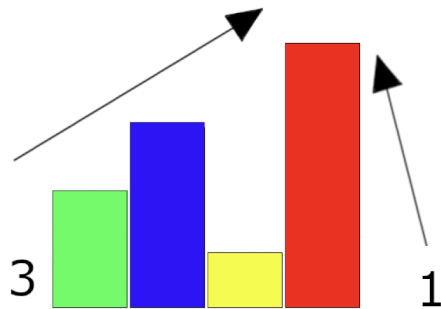
Önemli de olsa, film kültürü bu projeye yardım etmeyecek.

Chapter III

subject

	Exercise 00
	Rush-01
	Turn-in directory : <i>ex00/</i>
	Files to turn in : All necessary files
	Allowed functions : write , malloc , free

- Kaynak kodunuz şöyle derlenecektir: `gcc -Wall -Wextra -Werror -o rush-01 *.c`
- Gönderim dizininiz programınızın derlenmesi için gerekli olan bütün dosyaları içermelidir.
- Aşağıdaki sorunu çözen bir program oluşturun:
- 4x4 boyutlu bir haritada, mümkün olan her bakış açısından bütün sıra ve sütunların doğru sayıda kutuların görüneceği, her kareye 1'den 4'e kadar yükseklikte kutular yerleştirin. (satırlar için sol-sağ, sütunlar için aşağı-yukarı)
- Örnek: 3. yükseklikte bir kutu, 1. yükseklikte bir kutuyu soldan gizleyecektir, yani 3 görünür kutumuz olacaktır ve sağdan da sadece bir tane, çünkü 4. yükseklikte bir kutu her şeyi gizleyecektir.



- Her bir görüntüye (satır ve sütun başına 2'şer tane) bir değer verilecektir. Programınız, her bir satır ve sütunda her boyutta kutudan birer tane olacak şekilde, kutuları doğru şekilde yerleştirmelidir.
- Çıktımız, bulduğunuz ilk çözümü içerecektir.
- Programınızı şu şekilde başlatacağız :

```
> ./rush-01 "col1up col2up col3up col4up col1down col2down col3down col4down row1left row2left  
row3left row4left row1right row2right row3right row4right"
```

- (cf. annex 1)
- "col1up" sol sütunun üstten görüntüsü için tek değerdir. Her biri, '1' ve '4' arası bir karakterin değerler dizisini temsil eder.
- This is the ONLY acceptable input for your program. Any other input must be considered an error.
- Geçerli bir set için istenilen bir girdi/çıktı örneği:

```
./rush-01 "4 3 2 1 1 2 2 2 4 3 2 1 1 2 2 2" | cat -e  
1 2 3 4$  
2 3 4 1$  
3 4 1 2$  
4 1 2 3$
```

- (cf. annex 2 and 3)
- Hata durumunda, ya da herhangi bir çözüm bulamazsanız, yeni bir satır öncesi "Error" görüntüleyin.
- Bonus puan istiyorsanız, daha farklı bir harita boyutu işleyebilirsiniz. (En fazla 9x9 !!!!)
- Her zaman olduğu gibi, eğer bonus çalışıyorsa, ama zorunlu olan çalışmıyorsa, 0 alırsınız.

Chapter IV

Annex

Bu, programınızın artistik bir görüntüsüdür. Belli ki, bir önceki kısımda tarif edildiği gibi bir program göndermeniz gerekiyor.
Bu temsilin tek hedefi projeyi anlamana yardımcı olmaktır.

- Annex 1:

	col1up	col2up	col3up	col4up	
row1left					row1right
row2left					row2right
row3left					row3right
row4left					row4right
	col1down	col2down	col3down	col4down	

- Programınızın col_up, col_down, row_left ve row_right kullanarak temsili
- Annex 2:

	4	3	2	1	
4					1
3					2
2					2
1					2
	1	2	2	2	

- ol* ve row*'u değiştirerek bu sonucu elde ediyoruz.

- Annexe 3:

	4	3	2	1	
4	1	2	3	4	1
3	2	3	4	1	2
2	3	4	1	2	2
1	4	1	2	3	2
	1	2	2	2	

- Programınız ilk kısımda verilen kuralları kullanarak boşlukları doldurmalı.

Chapter V

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.