

# BLM1012 Yapısal Programlamaya Giriş Projesi\_v1

15 Mayıs'a kadar bu belgenin sonuna eklenecek olan versiyon güncellemeleri açıklamalarını kontrol ediniz

## TESLİM BİLGİLERİ:

>>> 30 Mayıs 2023 Salı Saat 23.59'a kadar, online.yildiz sistemine tanımlanan proje modülüne aşağıdaki kurallara göre teslim edilecektir .

>>> ÖğrenciNo.zip (içinde sadece ÖğrenciNo.c kod ve ÖğrenciNo.PDF rapor olacak)

>>> Google drive, onedrive vb bulut sistemlerinin linkleri ile online.yildiz.edu.tr sistemine proje yüklemesi kesinlikle kabul edilmeyecektir. Sadece dosya yüklemesi yapılmalıdır.

>>> Geç teslimler sistem kapandıktan sonra yapılamaz. Ayrıca mail ve diğer yollarla teslim mümkün değildir. Sisteme teslimin başarılı olarak yapıldığını gösteren ekran görüntüsünü kanıt olarak saklayınız. Kanıt olmaksızın yapılan itiraz ve mazeretler dikkate alınmayacaktır.

## PROGRAM VE RAPOR TASARIMINDA DİKKAT EDİLMESİ GEREKENLER:

Öncelikle aşağıdaki linkteki kodlama ile ilgili genel kuralları okuyunuz:

[https://docs.google.com/document/d/1ySbmyym0s3VGGVW\\_Zm5vN7o7N0ttxdgPZ4yorHt5z7E/edit?usp=sharing](https://docs.google.com/document/d/1ySbmyym0s3VGGVW_Zm5vN7o7N0ttxdgPZ4yorHt5z7E/edit?usp=sharing)

Bu kurallara ek olarak aşağıdaki kurallara özellikle dikkat ediniz:

- \*\* Her bir işlem Modüler tasarım mantığında yapılmalıdır. Bir başka ifadeyle, sadece main fonksiyonu içeren veya aynı işlemi tekrar eden kodları içeren çözümler geçersizdir
- \*\* ANSI C dilinde kodlama yapılmalıdır. Cpp dahil diğer dillerle yapılan çözümler geçersizdir.
- \*\* Kodunuzu DevC++ veya CodeBlocks ile derledikten sonra gönderiniz. Diğer derleyicilerin bulunduğu IDE'lerle yaptığınız çözümlerle elde ettiğiniz kod bizim tarafımızda çalışmazsa teslim edilen çözüm geçersizdir.
- \*\* Değişken ve fonksiyon isimlendirme, girinti ve anlaşılır kod yazma kurallarına dikkat ediniz. Tanımladığınız her fonksiyonun öncesinde fonksiyonun girdisi ve çıktısı hakkında açıklama satırı yazınız.
- \*\* Global ve static değişken, continue ve goto deyimlerini kullanmayınız. Break deyimini sadece ve sadece switch operasyonu için kullanabilirsiniz. Bu kuralı ihlal eden çözümler geçersizdir.
- \*\* Raporunuzda Kapak ve iletişim bilgileriniz, program görselleri, adım adım programın her bir çalışan kısmının ekran görüntüsü yer almalıdır. Ayrıca Manuel ve otomatik modda yol bulmak için tasarladığınız kodun Şekil-1'deki örnek veriye göre analizini göstermelisiniz.
- \*\* Raporda ekran görüntüsü verirken ekranda sadece ilgili kısmın görüntüsü veriniz. Anlaşılır ve açık olmayan ekranın tamamının görüntüsü puan kaybetmenize neden olabilir.

## PROJE HAKKINDA SORULARINIZ İÇİN DİKKAT EDİLMESİ GEREKENLER:

>>> C dilindeki kodlama ile ilgili sorun ve sorunlar için lab grup hocanızla irtibatlı olmalısınız

>>> Proje ile ilgili soruları 15 Mayıs 2023 tarihine kadar sadece classroom üzerinde açılacak olan soru sorma gönderisinin altına sormalısınız. Mail ile sorulan sorular dikkate alınmaz.

## PROJE KONUSU:

Matris ( $N \times N$ ) üzerinde bir sayı eşleştirme oyunu hazırlanacaktır. Örneğin  $N = 5$  için, 1 ile 5 arasındaki sayıların Şekil-1'deki matriste çiftler halinde yerleştirildiğini varsayalım (enkler önemsizdir). Geliştireceğiniz algoritmalarla bu matris üzerindeki sayıları aşağıdaki ayrıntılar bölümünde verilen kurallara göre eşleştiren yolları bulmanız gerekmektedir.

			2	5
	1			
			4	3
2	1	5		
4				3

Figure 1 Matrise yerleştirilen sayılar

## PROJENİN AYRINTILARI:

- Matris Oluşturma Kuralları:** Matrisin üzerine sayı yerleştirme oyunu matrisin boyutu olan  $N$  değerini kullanıcıdan alarak başlamalıdır. Bu  $N$  değerine göre iki farklı modda matrise sayılar yerleştirilmelidir. Boş hücre değeri olarak 0 veya -1 kullanabilirsiniz.
  - Rastgele Yazma Modu:** Girilen  $N$  değerine göre 1- $N$  arası sayıların her birisi ikişer defa matrise rastgele yerleştirilmelidir.
  - Dosyada Okuma Modu:** Size ek olarak verilen dosyalarla ve dosya okuma kodları ile sayılar matrise yerleştirilmelidir.

**Not:** 1b maddesinde verilen bazı örneklerin eşleştirilmiş halleri de sizlere ek olarak verilmiştir.

- Eşleştirme kuralları:** Numaralar aşağıdaki koşullara göre eşleştirilecektir.
  - Aynı sayılar birbiriyle eşleşmelidir.
  - Birbiriyle kesişmeyen  $N$  farklı yol elde edilmelidir. Örnek (Şekil-2'deki 5 yol)
  - Oyunu tamamlamak için boş hücreler kalmamalıdır.

Oyun tamamlandığında Şekil-2'deki gibi bir sonuç elde edilmelidir.

2	2	2	2	5
2	1	5	5	5
2	1	5	4	3
2	1	5	4	3
4	4	4	4	3

Figure 2 Tamamlanmış Oyun

3- Eşleştirme İçin Hamle Yapma Kuralları: Sayıları eşleştirme işleminde 2. Maddedeki kurallara ek olarak yapılacak hamleleri iki farklı yolla yapmanız gerekmektedir

- Otomatik Modda Eşleştirme:** Yazacağınız algoritma 1. ve 2. Maddedeki kurallara göre yolları otomatik olarak bulmalıdır. Bu işlem sırasında oluşturulan yollar, aşağıda açıklanan manuel modda olduğu gibi yapılan hataların geri alınması gibi işlemler ekrana raporlanmalı ve en sonunda oyun tamamlanınca elde edilen yolları gösteren matris Şekil-2'deki gibi ekrana yazdırılmalıdır.
- Manuel Modda Eşleştirme:** Eşleştirme işlemleri aşağıdaki örnekte gösterildiği gibi olmalıdır. Bu modda matris üzerinde indisleri verilen iki hücre arasındaki hücreleri 4 komşuluk (sağ, sol, yukarı, aşağı) durumuna göre doldurarak yollar oluşturulmalıdır.

**Örneğin:** Aşağıda oyunun sırasındaki bir fragmanda iki farklı 5 değerinin eşleştirilmesi adım adım gösterilmiştir.

- ADIM: Source: (3,2), Destination: (1,2)
- ADIM: Source: (1,2), Destination: (1,4)
- ADIM: Source: (1,4), Destination: (0,4) **Sayılar eşleştirildi**



**NOT:** Manuel ve otomatik oyun modunda bir işlem **YANLIŞ** yapıldıysa geri alınabilir (**UNDO**) şekilde algoritma geliştirilmeli ve **oyun kesinlikle sonlanmamalıdır**. Önceki örnekte olduğu gibi yapılan bir eşleştirme geri alındığında bir matrisin bir önceki durumuna geri gelinmelidir. Örnek bir durum aşağıda verilmiştir.



4- **Oyun Oynama Kuralları:** İlk 3 maddede verilen kurallara göre M sayıda kullanıcı için bu oyunlar oynanmalıdır. Her bir oyuncu istediği kadar oyunu oynamalı ve her bir oyunun skoru oyun sonunda kaydedilmelidir.

5- **Program Arayüzü:** Tasarlanan algoritmalarla oynanacak oyun için aşağıdaki kurallara göre program tasarımı yapılmalıdır. Bunlar dışında ayrıca eklentiler yapabilirsiniz

a. **Ana menü:**

- i. Rastgele Matris Oluştur:
- ii. Dosyadan Matris Oluştur:
- iii. Kullanıcıların Skorlarını Göster:
- iv. Çıkış

b. **Oyun Menüsü:** Ana menüde i ya da ii seçildiğinde, önce kullanıcı adı alınmalı ve aşağıdaki menülerden birinde Ana Menüye geri dönüş yapılana kadar oyun oynanmalıdır.

- i. Manuel Modda Oyna
- ii. Otomatik Modda Oyna
- iii. Ana Menüye Dön

6- **Kullanıcı Skorları:** Her bir kullanıcının her bir oyun oynadığında elde ettiği skoru hesaplayan bir formül geliştiriniz ve bu formüle göre kullanıcı skorlarını kaydediniz.

**Kullanıcı skoru hesabında aşağıdaki parametreleri içeren bir formül oluşturunuz**

- >> oyunun tamamlanma süresi,
- >> oyunun manuel/otomatik modda oynanma durumu,
- >> matrisin boyutu,
- >> matrisin rastgele/dosyadan oluşturulma şekli
- >> Yapılan yanlış işlem (UNDO) sayısı
- >> kullanıcının oynadığı oyun sayısı

7- **Algoritma karmaşıklığı:** Raporunuzda Manuel Mod ve Otomatik Mod için geliştirdiğiniz çözüm yönteminin zaman karmaşıklığı ifadesini  $T(n)$  olarak yazınız. Eğer bu ifadeler özyineleme içeriyorsa ifadenin çözümünü yapınız. Raporunuza sonucu yazınız.

8- **Proje Değerlendirme:** Projede rapor ve kod için ayrı ayrı 100 üzerinden puan verilecektir. Proje puanı =  $\%25 \cdot \text{rapor} + \%75 \cdot \text{kod}$  formülü ile hesaplanacaktır.

**Kod Değerlendirmesi:**

**Ana Menüyle İlgili Kodlar:** 15 puan

**Oyunu Manuel Oynama modülünün kodlanması:** 65 Puan

**Oyunu Otomatik Oynama modülünün kodlanması:** 20 Puan

**VERSİYON GÜNCELLEMELERİ:**

**VO : İlk metin**

V1 : 08/05/2023 tarihli güncelleme ile Algoritma Karmaşıklığı bölümü eklendi

ahmet — fsdfs 80