

TÜRKİYE CUMHURİYETİ
YILDIZ TEKNİK ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



ALGORİTMA ANALİZİ PROJE RAPORU

Öğrenci No: 19011038
Öğrenci Adı Soyadı: Şevval Bulburu
Öğrenci E-Posta: sevval.bulburu@std.yildiz.edu.tr

Ders/Grup: BLM3021 Algoritma Analizi/ 1.Grup

Ders Yürütücüsü
PROF DR MİNE ELİF KARSLIGİL
03.01.2023

YÖNTEM:

Algoritma Analizi proje ödevinde verilen $N \times N$ 'lik bir oyun tahtası içerisindeki her satırda N tane farklı renk bulunmaktadır. İstenen ise her sütunda her renkten yalnızca bir tane olacak şekilde oyun tahtasının düzenlenebiliyor ise düzenlenmesidir.

Problemin çözümünde string olarak ilk durumu kullanıcıdan alınan oyun tahtası string dizisinde tutulmaktadır. Çözümü kolaylaştırması adına string matris $N \times N$ 'lik bir integer matrisine dönüştürülmüştür. Dönüştürmede ilk satırın renk düzeni initial olarak alınmış ve sırasıyla renklere 1'den başlayarak N 'e kadar sayılar kodlanmıştır. Daha sonraki satırlar ise initial değerlere göre eşleştirilerek dönüşüm tamamlanmıştır.

Aynı satırdaki sıralamanın değişebilmesi için shiftMatrix() isimli bir fonksiyon tanımlanmıştır. Bu fonksiyon integer matrisi ve shift edilmesi gereken satır numarasını parametre olarak alarak $N - 1$. sütundaki renk kodunu 0. sütuna alarak diğer sütunları bir sağa kaydırmaktadır.

Bir sütunda kendinden önceki satırlarla aynı renk olup olmadığına karar verilmelidir. Eğer aynı renkler varsa ilgili sütun shift edilmelidir. Shift işlemi N . kez yapıldığında satır ilk haline döner. Bu yüzden maximum $N - 1$ shift işlemi yapılmalıdır. Bu durumu kontrol etmek ve gerçeklemek için check() fonksiyonu yazılmıştır. Bu fonksiyon integer matrisi, N değerini, kontrol edilecek satır numarasını ve her satırın shift sayılarını tutan shifts[] dizisini parametre olarak almaktadır. Fonksiyon recursive olarak yazılmış olup durma koşulu ilgili satırın shift sayısının $N - 1$ olmasıdır. Eğer bir satır $N - 1$ 'den küçük bir shift sayısında döndürülerek önceki satırlarla eşleşmeyen bir kombinasyon elde edilirse 1, elde edilemezse ve durma koşulu aşılsa 0 döndürmektedir. Bundan sonraki işlemler check() fonksiyonundan dönen değerlere göre yapılmaktadır.

Matris içerisinde satırlar arasında kombinasyonlar doğru sağlandıkça aşağı inilmeli, eğer bir satırda eşleşme sağlanamıyorsa bir üst satıra çıkılarak üst satırın başka bir kombinasyona getirilip getirilemeyeceği görülmelidir. Bu şekilde en fazla 0. Satıra kadar çıkılıp $N - 1$ satırına geri dönülebilir. Geri dönen her satır doğru kombinasyon ile sıralanmış ise çözüm bulunur. Herhangi bir satırda üstündeki satırlar maximum shift sayısına ulaştığı halde doğru kombinasyon sağlanamıyorsa bu oyun tahtası için çözüm yok demektir. Bu durumları kontrol eden fonksiyon organizeMatrix() fonksiyonudur. Recursive olarak yazılmıştır. Matrisi, N değerini, check fonksiyon sonucunu(flag), satır numarasını ve shifts[] dizisini parametre olarak almaktadır. check() fonksiyonundan dönen değer 1 ise bir alt satıra geçmekte, 0 ise bir üst satır maximum shift sayısından ($N - 1$ shift) daha az shift edilmişse bir üst satıra geçmektedir. Üst satıra geçilmediği durumda iki üstündeki satıra bakmaktadır ve bu durum 0. Satıra kadar devam etmektedir. Durma koşulları satır değerinin N 'e ulaşması veya -1 olmasıdır. Çözümün olduğu durum N 'e ulaşmasıdır ve organizeMatrix()'ten 1 döndürülür, olmadığı durum -1 durumudur ve organizeMatrix()'ten 0 döndürülerek mod fonksiyonlarına değerler aktarılmıştır.

Çözümler için normal ve detay mod için iki fonksiyon tanımlanarak print işlemleri için ilgili fonksiyonlara mode değeri de parametre olarak verilmiştir. Çözümün olduğu durumda integer matris tekrar string matrise çevrilerek iki oyun tahtası da kullanıcıya output olarak verilmiştir. Çözüm olmadığı durumda çözümün olmadığına dair bir output verilmiştir.

UYGULAMA:

- Normal Mod Çözümlü Matris:

4 * 4 matris kullanılmıştır.

```
C:\Users\sevva\OneDrive\Masaüstü\22-23\G_2\ALGORITMA_ANALİZ\Proje_190
Mode (Normal Mod -> 1 Detay Mod -> 2):
1
Oyun tahtasının boyutunu giriniz:
4
Oyun tahtasını giriniz:
mavi
sari
mor
kirmizi
mavi
sari
mor
kirmizi
mavi
sari
mor
kirmizi
mavi
sari
mor
kirmizi

Oyun tahtasi:
mavi sari mor kirmizi
mavi sari mor kirmizi
mavi sari mor kirmizi
mavi sari mor kirmizi

Integer Oyun tahtasi:
1 2 3 4
1 2 3 4
1 2 3 4
1 2 3 4

Cozumlenmis Oyun Tahtasi:

Integer Oyun tahtasi:
1 2 3 4
4 1 2 3
3 4 1 2
2 3 4 1

Oyun tahtasi:
mavi sari mor kirmizi
kirmizi mavi sari mor
mor kirmizi mavi sari
sari mor kirmizi mavi
```

- Detay Mod Çözümlü Matris:

4 * 4 matris kullanılmıştır.

```

C:\Users\sevva\OneDrive\Masaüstü\22-23\G...Z\ALGORITMA
Mode (Normal Mod -> 1 Detay Mod -> 2):
2
Oyun tahtasının boyutunu giriniz:
4
Oyun tahtasını giriniz:
mavi
sari
mor
kirmizi
mavi
sari
mor
kirmizi
mavi
sari
mor
kirmizi
mavi
sari
mor
kirmizi

Oyun tahtasi:
mavi sari mor kirmizi
mavi sari mor kirmizi
mavi sari mor kirmizi
mavi sari mor kirmizi

Integer Oyun tahtasi:
1 2 3 4
1 2 3 4
1 2 3 4
1 2 3 4

Integer Oyun tahtasi:
1 2 3 4
4 1 2 3
1 2 3 4
1 2 3 4
row: 1 shift: 1 match: 0

Integer Oyun tahtasi:
1 2 3 4
4 1 2 3
4 1 2 3
1 2 3 4
row: 2 shift: 1 match: 1

Integer Oyun tahtasi:
1 2 3 4
4 1 2 3
3 4 1 2
1 2 3 4
row: 2 shift: 2 match: 0

Integer Oyun tahtasi:
1 2 3 4
4 1 2 3
3 4 1 2
4 1 2 3
row: 3 shift: 1 match: 1

```

```

Integer Oyun tahtasi:
1 2 3 4
4 1 2 3
3 4 1 2
3 4 1 2
row: 3 shift: 2 match: 1

Integer Oyun tahtasi:
1 2 3 4
4 1 2 3
3 4 1 2
2 3 4 1
row: 3 shift: 3 match: 0

Cozumlenmis Oyun Tahtasi:

Integer Oyun tahtasi:
1 2 3 4
4 1 2 3
3 4 1 2
2 3 4 1

Oyun tahtasi:
mavi sari mor kirmizi
kirmizi mavi sari mor
mor kirmizi mavi sari
sari mor kirmizi mavi
Mode (Normal Mod -> 1 Detay Mod -> 2):

```

- Normal Mod Backtracking Yapılan Matris:

3*3 matris kullanılmıştır.

```
C:\Users\sewa\OneDrive\Masaüstü\22-23\G_2\ALGORITMA_ANALİZ\Proje_190110
Mode (Normal Mod -> 1 Detay Mod -> 2):
1
Oyun tahtasının boyutunu giriniz:
3
Oyun tahtasını giriniz:
mavi
sari
mor
sari
mor
mavi
mavi
mor
sari

Oyun tahtası:
mavi    sari    mor
sari    mor    mavi
mavi    mor    sari

Integer Oyun tahtası:
1        2        3
2        3        1
1        3        2
```

- Detay Mod Backtraking Yapılan Matris:

3*3 matris kullanılmıştır.

```
C:\Users\sevva\OneDrive\Masaüstü\22-23\G...Z\ALGORITMA
Mode (Normal Mod -> 1 Detay Mod -> 2):
2
Oyun tahtasının boyutunu giriniz:
3
Oyun tahtasını giriniz:
mavi
sari
mor
sari
mor
mavi
mavi
mor
sari

Oyun tahtasi:
mavi sari mor
sari mor mavi
mavi mor sari

Integer Oyun tahtasi:
1 2 3
2 3 1
1 3 2

Integer Oyun tahtasi:
1 2 3
1 2 3
1 3 2
row: 1 shift: 1 match: 1

Integer Oyun tahtasi:
1 2 3
3 1 2
1 3 2
row: 1 shift: 2 match: 0

Integer Oyun tahtasi:
1 2 3
3 1 2
2 1 3
row: 2 shift: 1 match: 1

Integer Oyun tahtasi:
1 2 3
3 1 2
3 2 1
row: 2 shift: 2 match: 1

Integer Oyun tahtasi:
3 1 2
3 1 2
3 2 1

Integer Oyun tahtasi:
3 1 2
2 3 1
3 2 1
row: 1 shift: 1 match: 0
```

```
Integer Oyun tahtasi:
3 1 2
2 3 1
1 3 2
row: 2 shift: 1 match: 1
```

```
Integer Oyun tahtasi:
3 1 2
2 3 1
2 1 3
row: 2 shift: 2 match: 1
```

```
Integer Oyun tahtasi:
3 1 2
1 2 3
2 1 3
row: 1 shift: 2 match: 0
```

```
Integer Oyun tahtasi:
3 1 2
1 2 3
3 2 1
row: 2 shift: 1 match: 1
```

```
Integer Oyun tahtasi:
3 1 2
1 2 3
1 3 2
row: 2 shift: 2 match: 1
```

```
Integer Oyun tahtasi:
2 3 1
1 2 3
1 3 2
```

```
Integer Oyun tahtasi:
2 3 1
3 1 2
1 3 2
row: 1 shift: 1 match: 0
```

```
Integer Oyun tahtasi:
2 3 1
3 1 2
2 1 3
row: 2 shift: 1 match: 1
```

```
Integer Oyun tahtasi:
2 3 1
3 1 2
3 2 1
row: 2 shift: 2 match: 1
```

```
Integer Oyun tahtasi:
2 3 1
2 3 1
3 2 1
row: 1 shift: 2 match: 1
```

```
Girilen Oyun Tahtasi Icin Cozum Yok.
Mode (Normal Mod -> 1 Detay Mod -> 2):
```

- Normal Mod Backtracking Yapılan Matris:

5*5 matris kullanılmıştır.

```
C:\Users\sevva\OneDrive\Masaüstü\22-23\G_2\ALGORITMA_ANALİZ\Proje
Mode (Normal Mod -> 1 Detay Mod -> 2):
1
Oyun tahtasının boyutunu giriniz:
5
Oyun tahtasını giriniz:
mavi
sari
mor
kirmizi
pembe
mavi
sari
pembe
mor
kirmizi
pembe
mavi
sari
mor
kirmizi
mavi
sari
mor
pembe
kirmizi
mavi
sari
mor
kirmizi
pembe

Oyun tahtası:
mavi    sari    mor    kirmizi pembe
mavi    sari    pembe mor    kirmizi
pembe   mavi    sari    mor    kirmizi
mavi    sari    mor    pembe   kirmizi
mavi    sari    mor    kirmizi pembe

Integer Oyun tahtası:
1      2      3      4      5
1      2      5      3      4
5      1      2      3      4
1      2      3      5      4
1      2      3      4      5

Girilen Oyun Tahtası İçin Çözüm Yok.
Mode (Normal Mod -> 1 Detay Mod -> 2):
```

SONUÇ:

Int – String çevirmeleri için kullanılan `convertInttoStr()` ve `convertStrToInt()` fonksiyonları için;

Yer karmaşıklığı -> $N * N$ matrisler kullanılmıştır. -> $O(N^2)$

Zaman karmaşıklığı -> $N * N$ kez döngüye girilir. -> $O(N^2)$

```
for(i = 0; i < N ; i++)
{
    for(j = 0; j < N; j++)
    {
        strcpy(result[k], colorMtrx[mtrx[i][j] - 1]);
        k++;
    }
}
```

Initial renk değerlerini almak için kullanılan `colorsMatrix()` fonksiyonu;

Yer karmaşıklığı -> $N * N$ matris kullanılmıştır. -> $O(N^2)$

Zaman karmaşıklığı -> N kez döngüye girilir. -> $O(N)$

```
for(i = 0; i < N; i++)
    strcpy(colorMtrx[i], mtrx[i]);
```

`organizeMatrix()` ve içerisinde kullanılan `check()`, `shiftMatrix()` fonksiyonları için;

Yer karmaşıklığı -> $N * N$ matrisler kullanılmıştır. -> $O(N^2)$

Zaman karmaşıklığı -> Her satır $N - 1$ kez shift edilebilir. -> $O(N^N)$

Maximum $N!$ kez backtracking ile satırlar dolaşılabilir.

-> $O(N^N)$

Maximum karmaşıklık `organizeMatrix()` içerisinde olduğu için algoritma karmaşıklığı da buna eşittir.

VIDEO LİNKİ:

https://drive.google.com/drive/folders/1mqko_Roe4vXnqxVdTfXfOtl7TuLtvYG0?usp=s_haring