

# 2020-2021 Güz Yarıyılı

## Algoritma Analizi

### Ödev – 1

#### Ders Yürütücüleri

Doç. Dr. M. Elif KARSLIĞİL  
Dr. Öğr. Üyesi M. Amaç GÜVENSAN

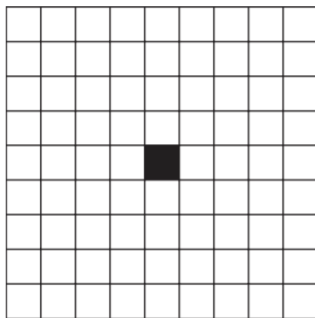
**Problem 1:** N elemanlı bir dizide birbirine en yakın değere sahip iki elemanın bulunması isteniyor.  
(40 Puan)

- Problemi brute-force\* yaklaşımı ile çözen algoritmanın C dilinde kodunu yazıp, verimliliğini analiz ediniz.
- Problemi brute-force yaklaşımından daha az karmaşıklığa sahip başka bir algoritmik yöntem ile çözüp analiz ederek verimliliğini ortaya koyunuz. Yeni yaklaşımınızı C dilinde kodlayınız.

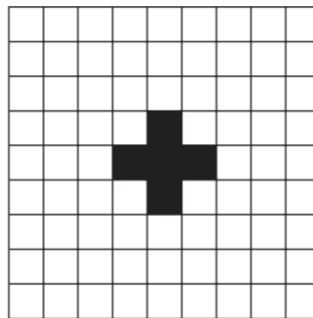
\* **Brute force** is a straightforward approach to solving a problem, usually directly based on the problem statement and definitions of the concepts involved.

**Problem 2:** Bir matriste, verilen N değeri dikkate alınarak, “Von Neumann’s Neighborhood” kuralına göre (<https://mathworld.wolfram.com/vonNeumannNeighborhood.html>) hücrelerin “1” değeri ile işaretlenmesi isteniyor. Aşağıda N=0,1,2 değerlerine göre matris içerisindeki hücreler siyah ile gösterilmektedir (Beyaz=“0”, Siyah = “1”).

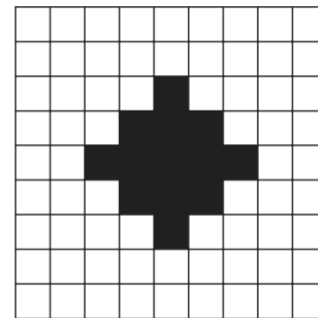
Von Neumann’s Neighborhood” kuralını gözeterek verilen N değerine göre matrisleri oluşturan programın kodunu C dilinde yazınız. Program çıktısı, ilgili matrisi ve satır başına ve toplam siyah hücre sayısını göstermelidir. Matris dinamik bellek ayırma yöntemi ile en uçtaki siyah hücrelerin dışında sadece bir satır ve sütun kalacak şekilde oluşturulmalıdır. Örnek N=0 için 3x3, N=1 için 5x5 boyutunda olmalıdır. Şekildeki matris boyutlarını DİKKATE ALMAYINIZ. (60 Puan)



$n = 0$



$n = 1$



$n = 2$

## **Ödev Teslimi ile ilgili önemli bilgiler:**

Aşağıda verilen bütün bilgileri içeren tek bir doküman hazırlayarak **07.11.2020 saat 23:59'a** kadar [online.yildiz.edu.tr](http://online.yildiz.edu.tr) adresi üzerinden **OgrenciNumarasi.rar** dosyasını yükleyiniz.

## **Teslim Edilecekler:**

1. Algoritmanızın **C** dilinde programını hazırlayarak dokümana da ekleyiniz.
2. Her algoritma için farklı durumlarını gösteren ekran çıktılarını dokümanda paylaşınız.
3. İlk soruda her iki şık için karmaşıklık analizini yapınız.
4. **Teslim Edilecekler**
  - a. HW#\_OgrenciNumarasi.rar (Örn: 15011001.rar)
    - i. OgrenciNumarasi\_QuestionNo.pdf (Örn: 15011001\_1.pdf)
    - ii. OgrenciNumarasi\_QuestionNo\_Part.c (Örn: 15011001\_1\_a.c)
    - iii. OgrenciNumarasi\_QuestionNo\_Part.c (Örn: 15011001\_1\_b.c)
    - iv. OgrenciNumarasi\_QuestionNo.pdf (Örn: 15011001\_2.pdf)
    - v. OgrenciNumarasi\_QuestionNo.pdf (Örn: 15011001\_2.c)

## **Değerlendirme**

### ***Algoritma Tasarımı ve Programın Çalışması: (%80)***

1. Ödev, istenilen işlerin tamamını yerine getirmelidir.
2. Gereksiz kontrollerden ve işlemlerden arınmış bir tasarım yapılmalıdır.
3. Program hatasız çalışmalıdır.
4. Programın çalışması sırasında, konuyu bilmeyen kişilerin rahatlıkla anlayabilmesi için, giriş ve çıkışlarda mesajlarla bilgi verilmelidir.

### ***Rapor Dokümantasyonu: (%20)***

1. Raporun kapak sayfasında, dersin adı, öğrencinin ad, soyad ve numarası, ödev konusu bilgileri yer almalıdır.
2. Kaynak kodda değişken deklarasyonu yapılırken her değişken tek satırda tanımlanmalı, tanımın yanına değişkenin ne için kullanılacağı açıklama olarak yazılmalıdır.
3. Değişken isimleri anlamlı olmalıdır.
4. Her fonksiyonun yaptığı iş, parametreleri ve dönüş değeri açıklanmalıdır.
5. Gerekli yerlerde açıklama satırları ile kodda yapılan işlemler açıklanmalıdır.
6. Gereksiz kod tekrarı olmamalıdır.
7. Kaynak kodun formatı düzgün, okunabilir ve takip edilebilir olmalıdır.