07.05.2019

VERİ YAPILARI VE ALGORİTMALAR 3.ÖDEV

Konu : Böl ve Yönet Algoritmalar

**Ad Soyad :** Yusuf ANI

**Öğrenci Numarası :** 16011033

**Grup :** 1 (MEK)

# İÇİNDEKİLER

[İÇİNDEKİLER 0](#_Toc8146343)

[YÖNTEM 2](#_Toc8146344)

[1.1 - Problem 2](#_Toc8146345)

[1.2-Akış 2](#_Toc8146346)

[UYGULAMA 3](#_Toc8146347)

[2.1 - Rastgele sayılardan oluşan diziler 3](#_Toc8146348)

[2.2 – Küçükten Büyüğe Sıralı Dizi 5](#_Toc8146349)

[2.3 – Büyükten Küçüğe Sıralı Dizi 6](#_Toc8146350)

[3 – Karmaşıklık Hesabı 7](#_Toc8146351)

[3.1-Best Case 7](#_Toc8146352)

[3.2-Worst Case 7](#_Toc8146353)

# 1-YÖNTEM

## 1.1 - Problem

***“Bir kulede bulunan N odayı açmak için farklı büyüklükte N adet anahtar gerekiyor.***

***Anahtarları diğer anahtarlarla veya kilitlerin büyüklüklerini diğer kilitlerle karşılaştırmadan, sadece***

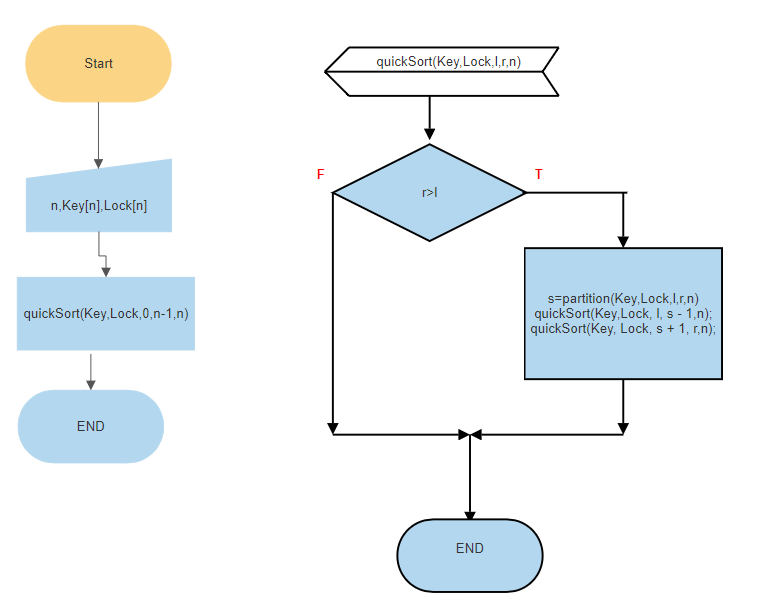
***anahtarlar kapılara takılarak kontrol yapılabiliyor. Her seferinde rasgele bir anahtar seçip hangi***

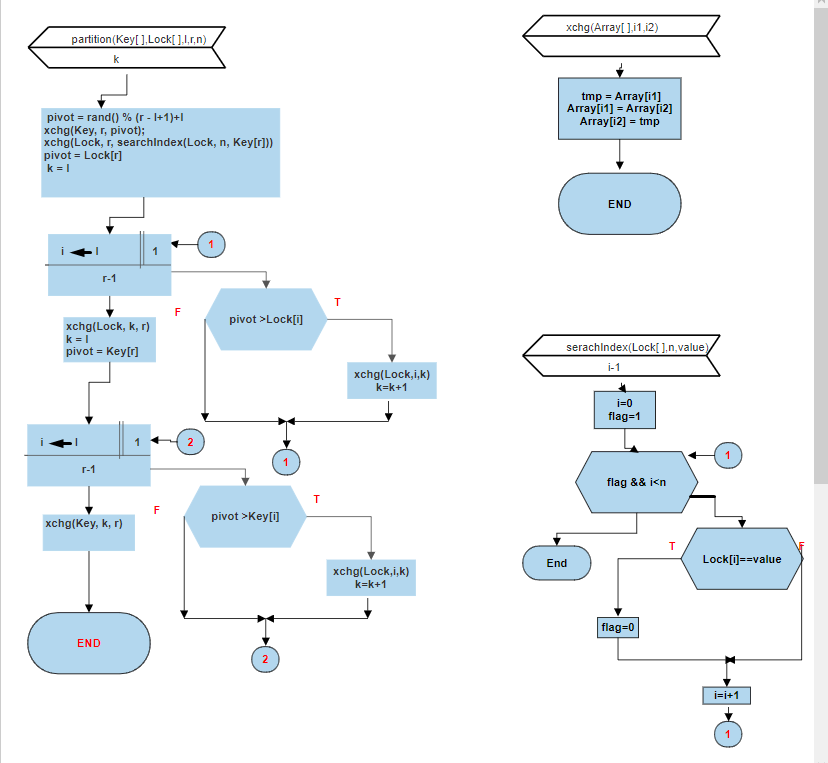
***kilide uyduğunu bularak, anahtarların uydukları bütün kapıları O(N\*lg2N) karmaşıklıkla bulan***

***algoritmayı tasarlayınız.”***

Probleme göre logaritmik olarak sıralayabileceğimiz bir algoritma istenmektedir. Veri Yapıları ve Algoritmalar Dersinde öğrendiğimiz quick sort algoritmasına random pivot seçimi eklenerek etkili bir şekilde bu problem çözülebilir.

## 1.2-Akış





# 2-UYGULAMA

## 2.1 - Rastgele sayılardan oluşan diziler

Key = { 6 , 2 , 5 , 1 , 3 , 4 }

Lock = { 5 , 1 , 3 , 6 , 4 , 2 }

## 2.2 – Küçükten Büyüğe Sıralı Dizi

Key = { 1 , 2 , 3 , 4 , 5 , 6}

Lock = { 1 , 2 , 3 , 4 , 5 ,6 }

## 2.3 – Büyükten Küçüğe Sıralı Dizi

Key = { 6 , 5 , 4 , 3 , 2 , 1}

Lock = { 6 , 5 , 4 , 3 , 2 , 1}

# 

# 3 –SONUÇ

3.1-Best Case :

Adım adım analiz işleminde olduğu gibi bir binary tree gibi düşünürsek Best case durumu her zaman pivotun ortadaki sayı olma durumudur . Bu durumda 2 tane dizi için 2n sıralama yapılır . O(logn\*2n) karmaşıklığa sahip olur .

3.2-Worst Case

En kötü durum diziyi bölme işleminin bize bir fayda getirmediği durumlar olabilir . Örnek olarak sıralı artan veya azalan dizi olabilir . 1,2,3,4 dizisi için önce 1 i seçip 2,3,4 sıralıcaz . Sonra 2’yi seçip sıralıcaz . Aynı şekilde 3 ve 4 için de durum aynı olacak . Bu yüzden bir tarafa doğru bir dallanmış ağaca sahip olacağız. Bu durumda n\*n ‘den O(n^2) karmaşıklığa sahip oluruz .